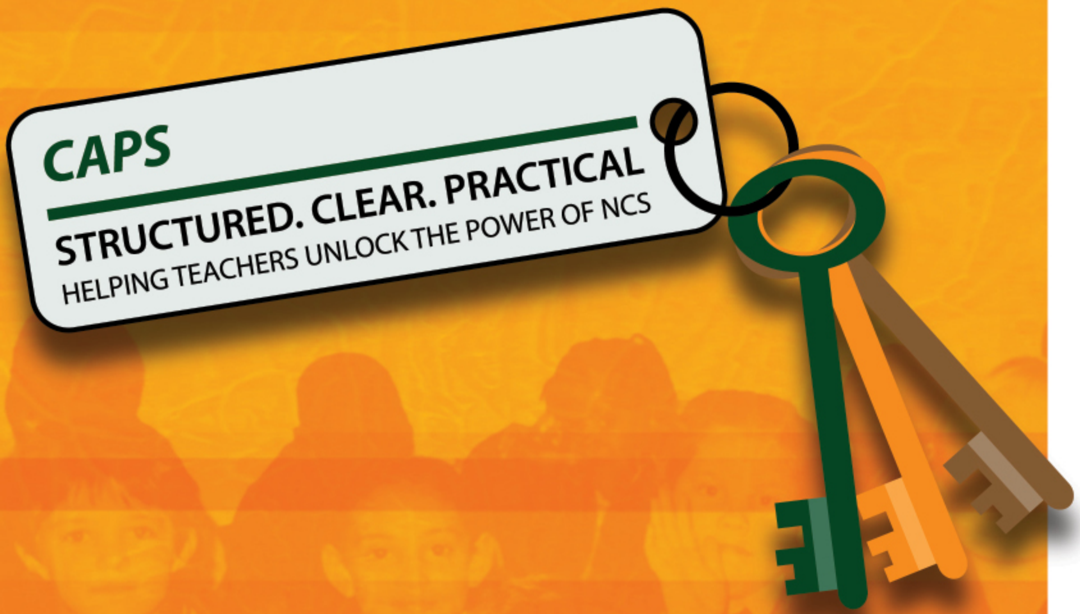
CODING AND ROBOTICS

National Curriculum Statement (NCS)

Curriculum and Assessment Policy Statement



FOUNDATION PHASE
GRADE R – 3

basic education
Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

Read to Lead
A Reading Nation is a Leading Nation

2030 NDP

30 YEARS OF FREEDOM

Department of Basic Education

222 Struben Street

Private Bag X895

Pretoria 0001

South Africa

Tel: +27 12 357 3000

Fax: +27 12 323 0601

120 Plein Street

Private Bag X9023

Cape Town 8000

South Africa

Tel: +27 21 465 1701

Fax: +27 21 461 8110

Website: http://www.education.gov.za

ISBN: 978-1-4315-3989-5

Design and Layout by: Department of Basic Education

Curriculum and Assessment Policy Statement (CAPS)

Coding and Robotics

Foundation Phase
Grade R – 3

FOREWORD BY THE MINISTER

In the last twenty-five years, our National Curriculum Statement (NCS) has been focused on transforming Education in South Africa. The democratic values enshrined in our Constitution (Act 108 of 1996) have inspired the development the National Curriculum. The Preamble to the Constitution states that the aims of the Constitution are to:

- *heal the divisions of the past and establish a society based on democratic values, social justice and fundamental human rights;*
- *improve the quality of life of all citizens and free the potential of each person;*
- *lay the foundations for a democratic and open society in which government is based on the will of the people and every citizen is equally protected by law; and*
- *build a united and democratic South Africa able to take its rightful place as a sovereign state in the family of nations.*

Education and the Curriculum have an important role to play in realising these aims. In 1997 Outcomes Based Education was introduced to overcome the Curricular divisions of the past and was reviewed in 2000. This led to the first Curriculum revision: the Revised National Curriculum Statement Grades R-9 and the National Curriculum Statement Grades 10-12 (2002).

In 2009 the Revised National Curriculum Statement (2002) was revised due to implementation challenges. The National Curriculum Statement Grade R-12 was developed in 2012 which combined the 2002 Curricula for Grade R-9 and Grades 10-12. The National Curriculum Statement for Grades R-12 builds on the previous curriculum but also updates it and aims to provide clearer specification of what is to be taught and learnt on a term-by-term basis.

The Curriculum has been developed encompassing the vision of the National Development Plan (NDP) aligning the Skills, Knowledge and Values required for the Technological Developments in the workplace. The NDP goals are aligned to the Sustainable Development Goals (SDG) and the African Union Agenda 2063. The Modern workplace requires learners that can adapt to a fast-changing home and work environments through empowering learners with the skills they develop through the Three Stream Model. These goals will be achieved through Differentiated Pathways and Multi-Certification levels.

The National Curriculum Statement Grades R-12 accordingly replaces the Subject Statements, Learning Programme Guidelines and Subject Assessment Guidelines with the

a) *Curriculum and Assessment Policy Statements (CAPS) for all approved subjects listed in this document;*
b) *National Policy Pertaining to the Programme and Promotion requirements of the National*
c) *Curriculum Statement Grades R-12 (N4PR Revised); and*
d) *National Protocol for Assessment Grades R-12 (NPA).*

Mrs Angie Motshekga,
MP Minister of Basic Education

CONTENTS

TABLES AND FIGURES

Tables

Figures

1 SECTION 1
INTRODUCTION TO THE CURRICULUM AND ASSESSMENT POLICY STATEMENT FOR CODING AND ROBOTICS FOUNDATION PHASE (GRADE R – 3)

1.1 BACKGROUND

The *National Curriculum Statement Grades R – 12 (NCS)* stipulates policy on curriculum and assessment in the schooling sector.

To improve implementation, the National Curriculum Statement was amended, with the amendments coming into effect in January 2012. A single comprehensive Curriculum and Assessment Policy document was developed for each subject to replace Subject Statements, Learning Programme Guidelines and Subject Assessment Guidelines in Grades R - 12.

1.2 OVERVIEW

(a) The *National Curriculum Statement Grades R – 12 (January 2012)* represents a policy statement for learning and teaching in South African schools and comprises the following:

   (i) National Curriculum and Assessment Policy Statements for each approved school subject*;*

   (ii) The policy document, National policy pertaining to the programme and promotion requirements of the National Curriculum Statement Grades R – 12; and

   (iii) The policy document, National Protocol for Assessment Grades R – 12 (January 2012).

(b) The *National Curriculum Statement Grades R – 12 (January 2012)* replaces the two current national curricula statements, namely the

   (i) *Revised National Curriculum Statement Grades R - 9, Government Gazette No. 23406* of 31 May 2002*,* and

   (ii) *National Curriculum Statement Grades 10 - 12 Government Gazettes, No. 25545 of 6 October 2003* and *No. 27594* of 17 May 2005.

(c) The national curriculum statements contemplated in subparagraphs (a) and (b) comprise the following policy documents which will be incrementally repealed by the *National Curriculum Statement Grades R – 12 (January 2012)* during the period 2012-2014:

   (i) The Learning Area/Subject Statements, Learning Programme Guidelines and Subject Assessment Guidelines for Grades R - 9 and Grades 10 – 12;

   (ii) The policy document, *National Policy on assessment and qualifications for schools in the General Education and Training Band d*, promulgated in *Government Notice No. 124* in *Government Gazette No. 29626* of 12 February 2007;

   (iii) The policy document, the *National Senior Certificate: A qualification at Level 4 on the National Qualifications Framework (NQF)*, promulgated in *Government Gazette No.27819* of 20 July 2005;

   (iv) The policy document, *An addendum to the policy document, the National Senior Certificate: A qualification at Level 4 on the National Qualifications Framework (NQF), regarding learners with special needs,* published in *Government Gazette, No.29466* of 11 December 2006, is incorporated in the policy document, *National policy pertaining to the programme and promotion requirements of the National Curriculum Statement Grades R – 12; and*

   (v) The policy document, *An addendum to the policy document, the National Senior Certificate: A qualification at Level 4 on the National Qualifications Framework (NQF), regarding the National Protocol for Assessment (Grades R – 12)*, promulgated in *Government Notice No.1267* in *Government Gazette No. 29467* of 11 December 2006.

(c) The policy document, *National policy pertaining to the programme and promotion requirements of the National Curriculum Statement Grades R – 12*, and the sections on the Curriculum and Assessment Policy as contemplated in Chapters 2, 3 and 4 of this document, constitute the norms and standards of the *National Curriculum Statement Grades R – 12.* It will therefore, in terms of *section 6A* of the *South African*

Schools Act, 1996 (Act No. 84 of 1996,) form the basis for the Minister of Basic Education to determine minimum outcomes and standards, as well as the processes and procedures for the assessment of learner achievement to be applicable to public and independent schools.

1.3 General aims of the South African Curriculum

- The *National Curriculum Statement Grades R - 12* gives expression to the knowledge, skills and values worth learning in South African schools. This curriculum aims to ensure that children acquire and apply knowledge and skills in ways that are meaningful to their own lives. In this regard, the curriculum promotes knowledge in local contexts, while being sensitive to global imperatives.
- The National Curriculum Statement Grades R - 12 serves the purposes of:
  - equipping learners, irrespective of their socio-economic background, race, gender, physical ability or intellectual ability, with the knowledge, skills and values necessary for self-fulfilment, and meaningful participation in society as citizens of a free country;
  - providing access to higher education;
  - facilitating the transition of learners from education institutions to the workplace; and
  - providing employers with a sufficient profile of a learner's competences.
- The National Curriculum Statement Grades R - 12 is based on the following principles:
  - Social transformation: ensuring that the educational imbalances of the past are redressed, and that equal educational opportunities are provided for all sections of the population;
  - Active and critical learning: encouraging an active and critical approach to learning, rather than rote and uncritical learning of given truths;
  - High knowledge and high skills: the minimum standards of knowledge and skills to be achieved at each grade are specified and set high, achievable standards in all subjects;
  - Progression: content and context of each grade shows progression from simple to complex;
  - Human rights, inclusivity, environmental and social justice: infusing the principles and practices of social and environmental justice and human rights as defined in the Constitution of the Republic of South Africa. The National Curriculum Statement Grades R – 12 is sensitive to issues of diversity such as poverty, inequality, race, gender, language, age, disability and other factors;
  - Valuing indigenous knowledge systems: acknowledging the rich history and heritage of this country as important contributors to nurturing the values contained in the Constitution; and
  - Credibility, quality and efficiency: providing an education that is comparable in quality, breadth and depth to those of other countries.
- The National Curriculum Statement Grades R - 12 aims to produce learners that can:
  - identify and solve problems and make decisions using critical and creative thinking;
  - work effectively as individuals and with others as members of a team;
  - organise and manage themselves and their activities responsibly and effectively;
  - collect, analyse, organise and critically evaluate information;
  - communicate effectively using visual, symbolic and/or language skills in various modes;
  - use science and technology effectively and critically showing responsibility towards the environment and the health of others; and
  - demonstrate an understanding of the world as a set of related systems by recognising that problem solving contexts do not exist in isolation.
- Inclusivity should become a central part of the organisation, planning and teaching at each school. This can only happen if all teachers have a sound understanding of how to recognise and address barriers to learning, and how to plan for diversity.

The key to managing inclusivity is ensuring that barriers are identified and addressed by all the relevant support structures within the school community, including teachers, District-Based Support Teams, Institutional-Level Support Teams, parents and Special Schools as Resource Centres. To address barriers in the classroom, teachers should use various curriculum differentiation strategies such as those included in the Department of Basic Education's *Guidelines for Inclusive Teaching and Learning* (2010).

1.4 TIME ALLOCATION

1.4.1 Foundation Phase

(a) The instructional time in the Foundation Phase is as follows:

Subject	Grade R (Hours)	Grades 1-2 (Hours)	Grade 3 (Hours)
Home Language	10	7/8	7/8
First Additional Language		2/3	3/4
Mathematics	7	7	7
Life Skills	**5**	**5**	**5**
• Beginning Knowledge	(1)	(1)	(1,5)
• Creative Arts	(1,5)	(1,5)	(1,5)
• Physical Education	(1,5)	(1,5)	(1)
• Personal and Social Well-being	(1)	(1)	(1)
Coding and Robotics	(1)	(1)	(2)
Total	**23**	**23**	**25**

(b) Instructional time for Grades R, 1 and 2 is 23 hours and for Grade 3 is 25 hours.

(c) Ten hours are allocated for languages in Grades R-2 and 11 hours in Grade 3. A maximum of 8 hours and a minimum of 7 hours are allocated for Home Language and a minimum of 2 hours and a maximum of 3 hours for Additional Language in Grades R – 2. In Grade 3 a maximum of 8 hours and a minimum of 7 hours are allocated for Home Language and a minimum of 3 hours and a maximum of 4 hours for First Additional Language.

(d) In Life Skills Beginning Knowledge is allocated 1 hour in Grades R – 2 and 2 hours as indicated by the hours in brackets for Grade 3.

1.4.2 Intermediate Phase

The instructional time in the Intermediate Phase is as follows:

Subject	Hours
Home Language	6
First Additional Language	5
Mathematics	6
Natural Sciences	2,5
Social Sciences	3
Life Skills	**3**
• Creative Arts	(1)
• Physical Education	(1)
• Personal and Social Well-being	(1)
Coding and Robotics	2
Total	**27,5**

1.4.3 Senior Phase

(a) The instructional time in the Senior Phase is as follows:

Subject Choice: Option 1	Subject Choice: Option 2	Hours
Home Language	Home Language	5
First Additional Language	First Additional Language	4
Mathematics	Mathematics	4,5
Natural Science	Natural Science	3
Social Sciences	Social Sciences	3
*Technology	*Economic Management Sciences	2
Coding and Robotics	Coding and Robotics	2
Life Orientation	Life Orientation	2
Arts and Culture	Arts and Culture	2
Total		**27,5**

* Schools/Learners can follow Option 1 (MST Stream) or Option 2 (Business Stream)

1.4.4 Grades 10-12

(a) The instructional time in Grades 10-12 is as follows:

Subject	Time allocation per week (hours)
I. Home Language	**4.5**
II. First Additional Language	**4.5**
III. Mathematics	**4.5**
IV. Life Orientation	**2**
V. A minimum of any three subjects selected from Group B Annexure B, Tables B1-B8 of the policy document, *National policy pertaining to the programme and promotion requirements of the National Curriculum Statement Grades R – 12,* subject to the provisos stipulated in paragraph 28 of the said policy document.	**12 (3x4h)**

The allocated time per week may be utilised only for the minimum required NCS subjects as specified above and may not be used for any additional subjects added to the list of minimum subjects. Should a learner wish to offer additional subjects, additional time must be allocated for the offering of these subjects.

2 SECTION 2: DEFINITION, AIMS, SKILLS AND CONTENT

2.1 INTRODUCTION

Coding and Robotics represents an interdisciplinary and multidisciplinary subject that integrates various components of STEAM (Science (including Computer Science), Technology, Engineering, Arts, and Mathematics).

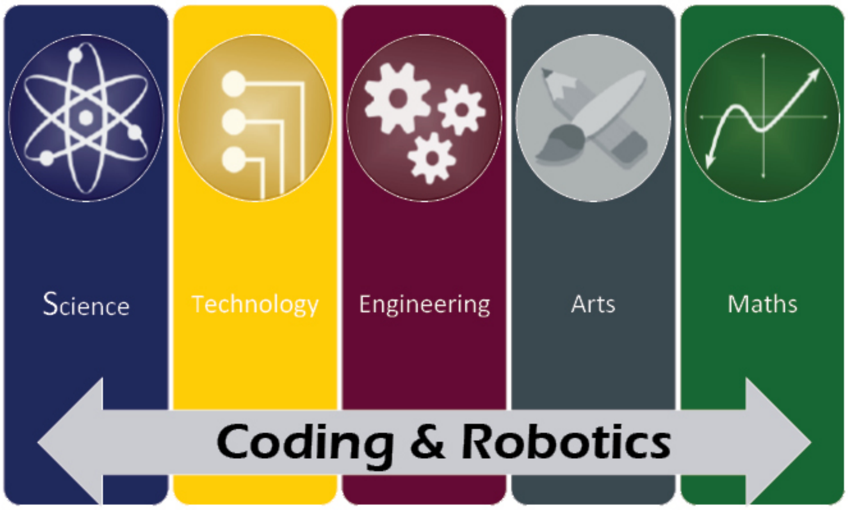


Figure 2.1 Coding and robotics as a STEAM discipline

The main driving force behind the uptake and surge of Coding and Robotics as a subject at school level is the link to the 4th and 5th industrial revolution (4IR, and 5IR). In the context of this curriculum the focus resides in the grounding concepts of STEAM related subjects.

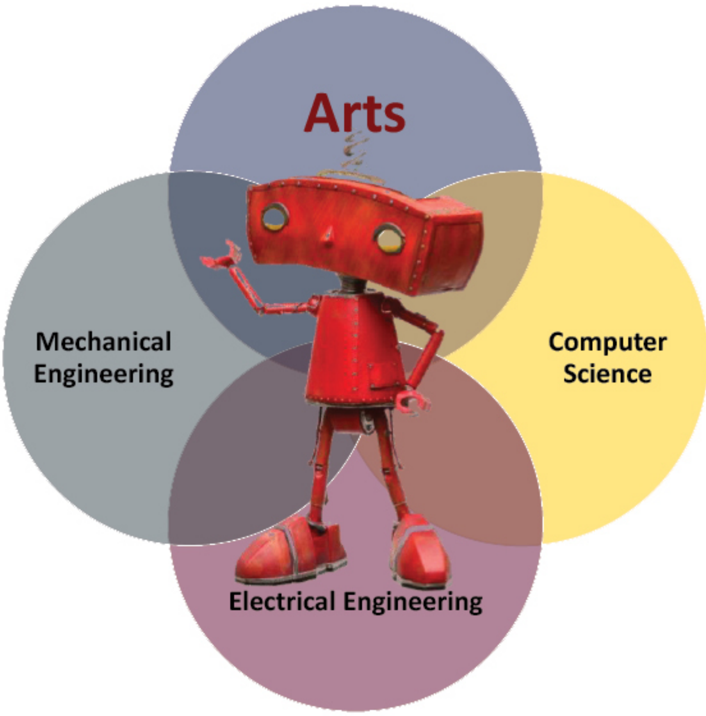


Figure 2.2: Coding and Robotics as a multi-disciplinary subject

2.2 WHAT IS CODING AND ROBOTICS

Coding and robotics combine the principles of programming with the design, construction, and operation of robots. Programming concepts, practices, and perspectives are applied to control robots to perform specific tasks. It includes digital concepts that refer to various ideas, principles and processes that are associated with digital technologies and their use.

The Coding and Robotics curriculum is based on the following pillars as depicted in the figure below.

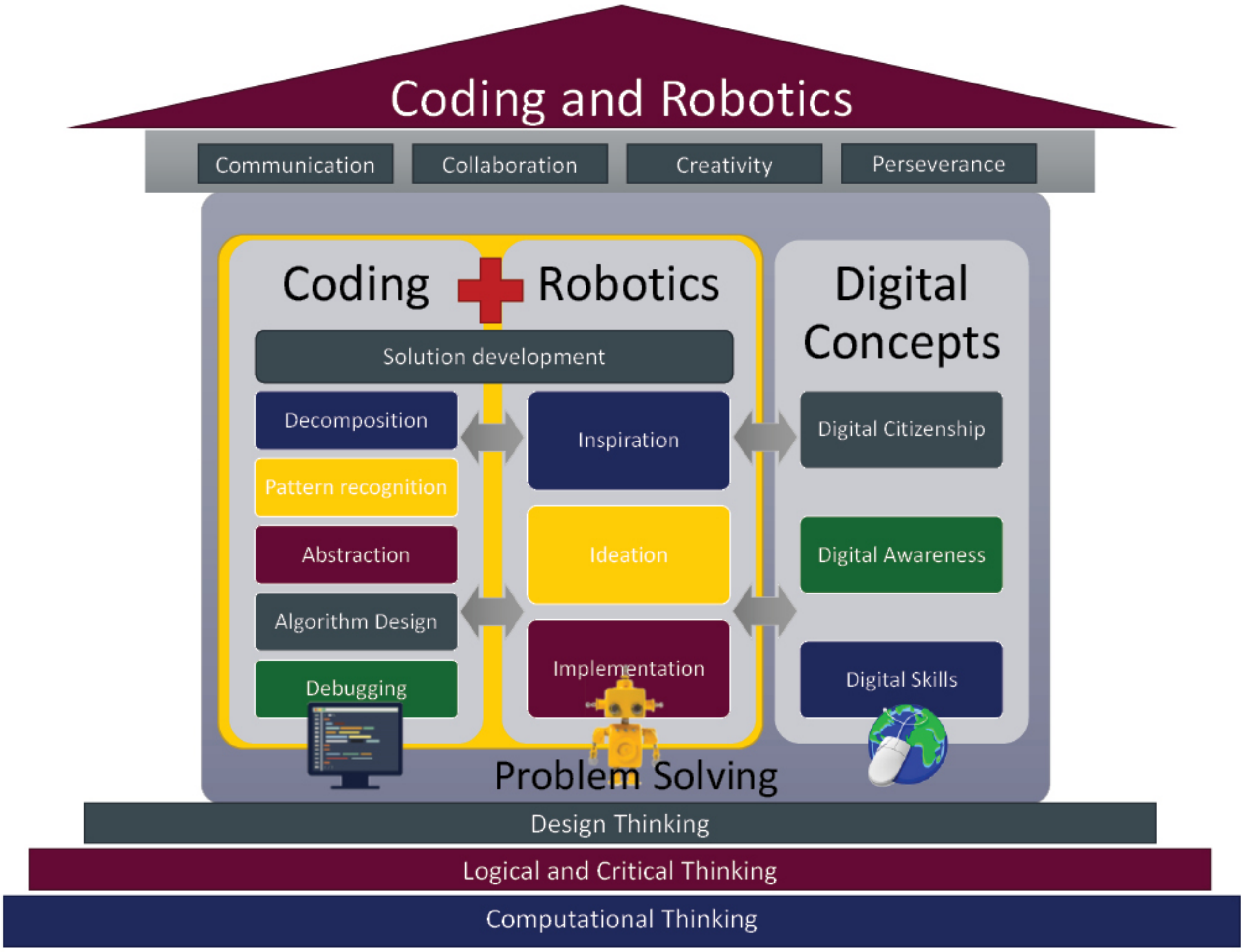


Figure 2.3: Overview of Coding and Robotics as a Subject

Coding is the process of creating a logical set of instructions that a human or computing device can understand and execute, which require a deep understanding of computational thinking and problem solving.

Robotics deals with the design, operation, and use of robots that can be programmed to perform tasks autonomously or semi-autonomously or by direct control. It presents the learners with the opportunity to see their thinking, design, and code in action.

Digital concepts encompass a range of digital literacy skills and awareness that enables learners to leverage digital technologies to their fullest potential and use digital tools responsibly.

2.3 SPECIFIC AIMS

The teaching and learning of Coding and Robotics (C&R) aim to develop the following for the learner to be able to:

- develop computational thinking skills to solve problems.
- advance design thinking to develop creative and human-centred approaches to solve problems.
- become part of a generation of creative, innovative systems thinkers that can use coding, robotics, and digital competencies to express their ideas.

- foster creativity, critical thinking, collaboration, communication, and innovation.
- function ethically and effectively in a digital and information-driven world.
- develop a critical awareness of how technologies impact society at large.
- instil self-efficacy and confidence to deal with situations requiring computational thinking, design thinking and problem solving.
- prepare for future careers in STEAM related fields.
- adopt a culture of being self-directed, life-long learners who can apply their skills in a wide range of contexts and situations (adaptable, flexible and resilient).

2.4 SPECIFIC SKILLS

The following skills are specifically emphasised:

2.4.1 Computational Thinking

Computational thinking is an attitude and a skill set where one uses specific techniques and strategies to complete tasks successfully and to solve problems systematically. It further helps one in arriving at a solution that both humans and a computer can understand.

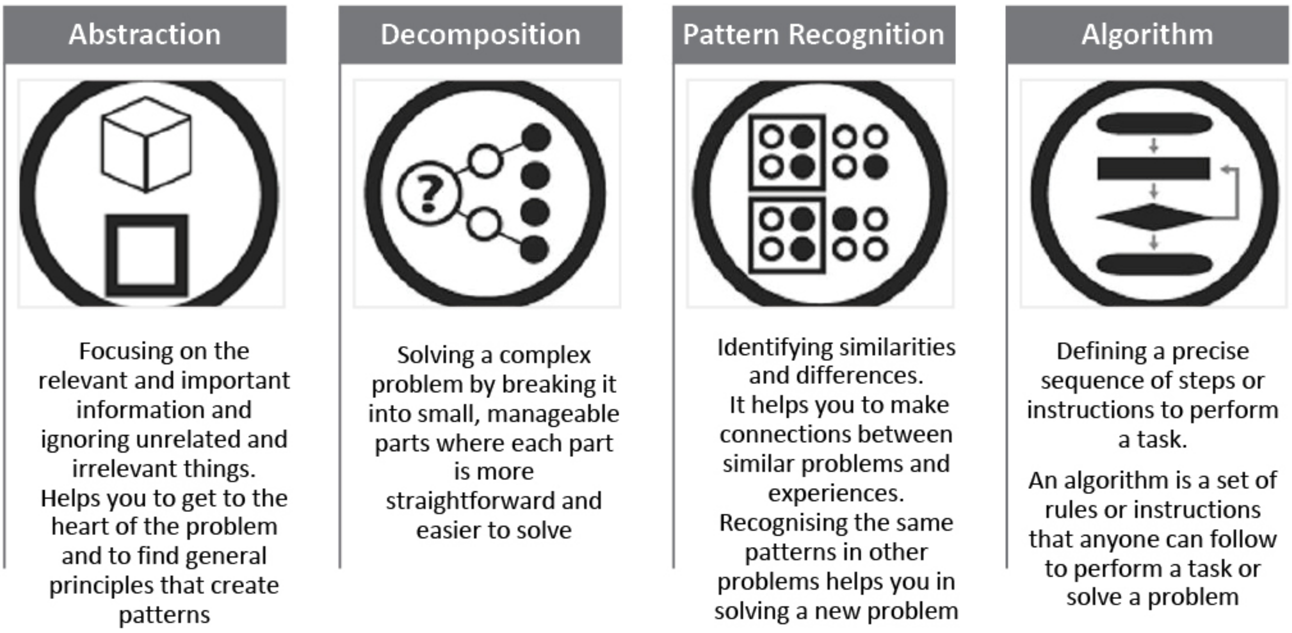


Figure 2.4: Computational Thinking Pillars

In Coding and Robotics, computational thinking helps learners to develop problem-solving strategies which they can apply when developing coding solutions (algorithms) as well as robotics solutions. It can also be applied to solve everyday life.

In terms of robotics, learners are demonstrating computational thinking concepts and practices when designing, constructing, and programming a robot. The robot's performance demonstrates the result of the learner's computational thinking practices as they iteratively test and debug their coding.

2.4.2 Design Thinking

In education, design thinking (DT) refers to a human-centred approach that encourages creativity and innovation when generating user-focused products, services, or experiences. DT is often expressed as an activity that involves the three **I**s process, namely:

- **Inspiration:** where creative thinking is applied to tackle a problem or challenge at hand, by gaining a deeper understanding of the problem and its context as well as to identify opportunities for innovation.
- **Ideation:** involves the generation of a wide range of ideas and potential solutions using various approaches such as brainstorming, prototyping and experimentation.
- **Implementation:** where the ideas and potential solutions are put into action. It includes testing, getting feedback and subsequent improvements of the design or solution.

Related to the three **I**s, is the notion that Design Thinking is also a problem-solving approach that combines creativity with structure and human-centred methods to understand and tackle challenges which involves empathizing with users, defining their needs, ideating possible solutions, prototyping, and testing those solutions, and iterating based on feedback. The following describes the design process:

- **Empathise:** involves gaining an understanding of who the end user is in a specific context, and how the envisaged solution will be appropriate towards addressing the problem.
- **Define:** relates to specifying in detail what the users' needs are, which could include the goals, skills available, and core principles that will guide the work to be done.
- **Ideate:** pertains to the creation of ideas and solutions using techniques such as brainstorming.
- **Prototype:** concerns the creation of one or several solutions to address the problem at hand.
- **Test:** relates to the process of determining how well the solution solves or address the problem. In this phase, feedback is important as the feedback could be used towards the improvement and enhancement and/or redesign of the complete solution or artefact.

Figure 2.6 depicts the relationship between the Design Thinking and Design Problem Solving approach.

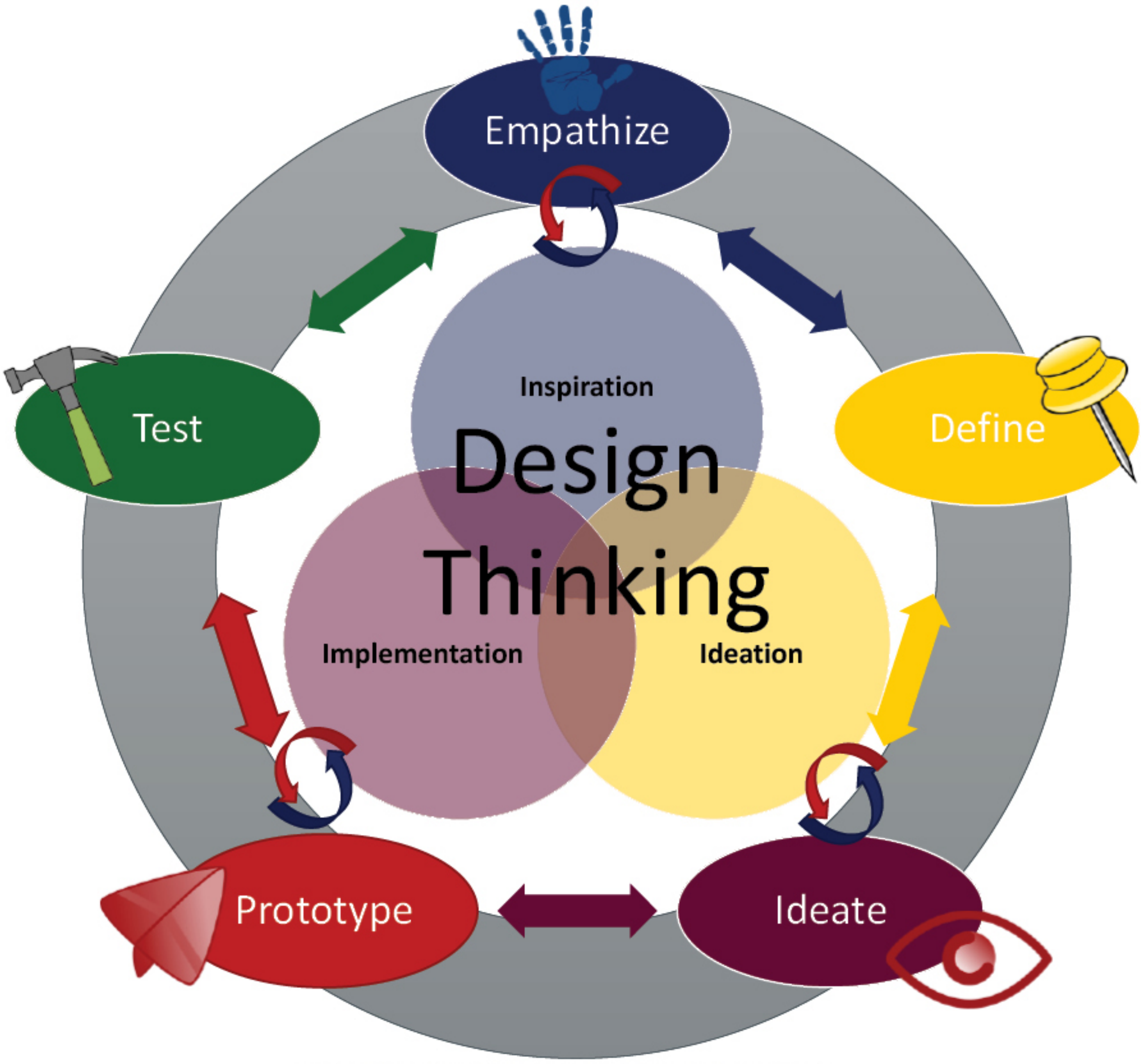


Figure 2 5: Design Thinking and Problem Solving Process

2.5 High-Level Competencies – Coding and Robotics

The three main topical areas of coding and robotics each comprises a set of key learning competencies central to their area of focus.

The following diagram outlines the three main topical areas and the main learning competencies associated with each at the final stage of curriculum that the learner should show proficiency at the appropriate level.

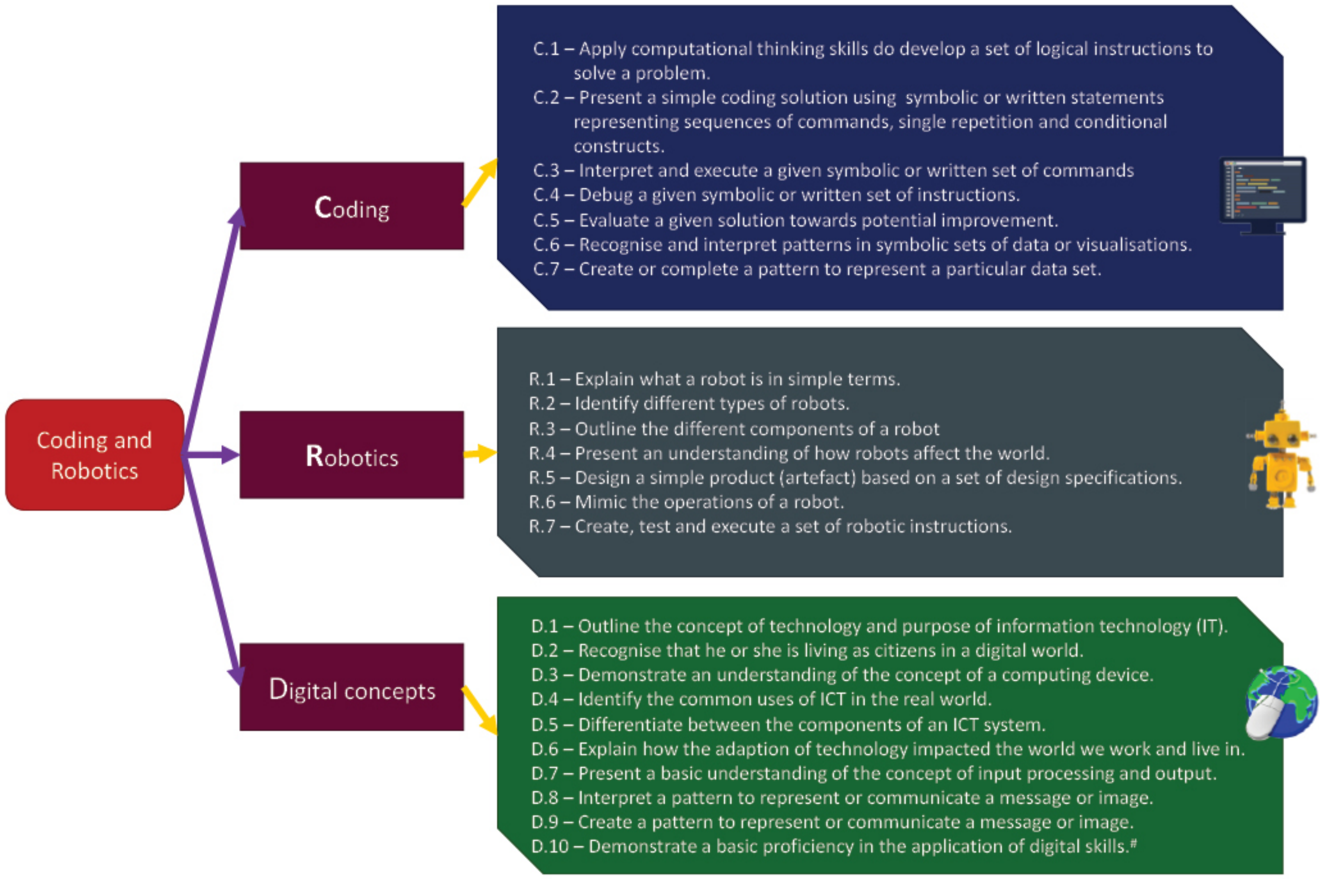


Figure 2.6 High-level Curriculum Competencies

A competence is a combination of knowledge skills, attitudes, and values which is reflected in behaviour that can be observed, measured, and evaluated. It refers to the ability to perform a specific task successfully and efficiently or in a manner that yields desirable outcomes.

2.6 Coding and Robotics Concepts, Practices and Perspectives

2.6.1 Coding

In coding, the following concepts, practices, and perspectives must be developed and practised repeatedly:

Concepts	Practices	Perspectives
• Algorithm • Sequence • Loop (Iteration) • Conditional (Decisions) • Operator • Logic • Data • Event • Debug • Representation • Parallelism • Automation	• Abstraction • Decomposition • Pattern Recognition • Generalisation • Algorithm Design • Incremental Development • Testing and Debugging • Evaluation • Modularise • Logical thinking • Creating computational artefacts	• Expressing and Creating • Questioning • Connecting • Collaboration • Perseverance • Choice of Conduct

Figure2.7 Coding Concepts, Practices and Perspectives

2.6.2 Robotics

In addition to the coding concepts, practices and perspectives, in robotics, the following concepts, practices, and perspectives must be developed and practised repeatedly:

Concepts	Practices	Perspectives
• Motion • Sensor • Actuator • Controller • Logic • Power Source • Automation • Instruction • Communication • Coding (Programming)	• Computational Thinking • Design Thinking • Prototyping • Design and Construction • Algorithm Design • Testing and Reconfiguration • Reflection and Iteration • Creative Thinking • Logical thinking • Creating robotics artefacts	• Expressing and Creating • Innovation • Questioning • Connecting • Collaboration • Perseverance • Choice of Conduct

Figure 2.8 Robotics Concepts, Practices and Perspectives

2.6.3 Digital Concepts

Digital concepts are fundamental ideas and principles that underpin and support coding and robotics. They encompass various aspects of technology and computer science, providing the context and application for these fields. In Coding and Robotics, digital concepts are divided into the following topics: Digital Citizenship, Digital Awareness and Digital Skills.

2.6.3.1 Digital Citizenship

Digital Citizenship helps to develop an awareness of responsible and ethical behaviour in the digital world, which includes the responsible and ethical use of digital tools.

It includes the rights, responsibilities, and behaviours (respect, integrity, and safety) of individuals in the digital world. It further includes concepts like respecting others' privacy, avoiding cyberbullying, netiquette, digital health, and welfare, as well as the impact and responsibility of online actions and deeds.

2.6.3.2 Digital Awareness

The recognition of the competencies, expertise, and mindset necessary for individuals to effectively use digital tools. This entails understanding and applying technology in a world that is becoming more interconnected.

It includes an awareness of different types of computing devices and their purpose concepts of hardware and software as well as concepts of input-processing-output, awareness of the internet as an example of a network and that, in a digital world, devices often need to communicate with each other.

2.6.3.3 Digital Skills

An essential set of a range of abilities that enable individuals to effectively use digital devices, software, and platforms to perform various tasks. It includes an awareness of patterns to communicate a message as patterns is a fundamental concept in both coding and robotics.

2.7 APPROACH TO TEACHING CODING AND ROBOTICS

Coding and Robotics, as a subject, is process-driven as it focuses on coding and robotics processes, rather than just exit skills or products. Coding develops cognitive and critical thinking skills as it emphasises the development of knowledge, skills, strategies, and attitudes that enable learners to become more effective individuals. Coding and Robots also supports learners to develop metacognitive skills, which include planning, developing, testing, evaluation and reflecting.

2.7.1 Problem-based Learning

Teaching and learning will follow a problem-based learning approach. Problem-based learning (PBL) is an active and learner-centred approach to learning involving several cognitive processes that aims to develop critical thinking, problem-solving, and collaboration skills. The goal of PBL is to help learners learn how to apply knowledge and skills in real-life situations, rather than just memorising information for tests. PBL also encourages learners to ask questions and seek answers, rather than passively receiving information. It also supports the development of self-directed learning.

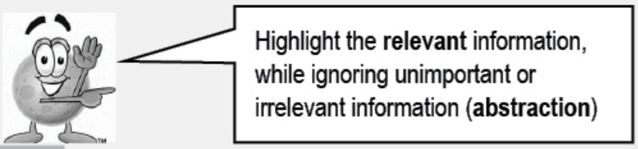
In Foundation Phase, learners will be given small, manageable problems which they need to solve using a problem-solving process. To develop and enhance self-efficacy (the learner's belief that he/she will be able to complete the task or solve the problem), the challenge of the task or problem should match the learner's competencies.

Example of a manageable problem and algorithm development using the problem-solving process in Foundation Phase:

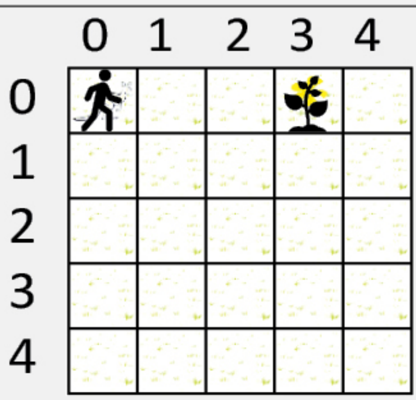
Problem:
Katlego needs to replant a flower in a different position (see diagram below).

Step 1: Understand the problem.

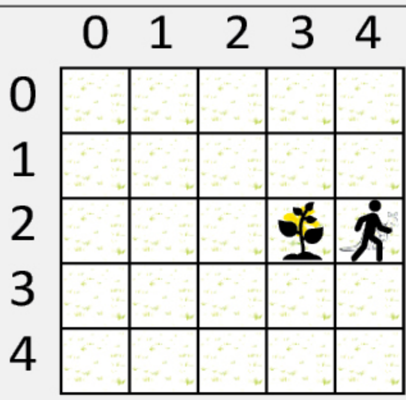
Katlego starts at (0, 0) facing right (East) (towards a flower) with no flowers in his hand. There is a flower at location (3, 0).
Develop instructions that will direct Katlego to pick the flower and plant it at location (3, 2).
After planting the flower, Katlego should move one space to the right (East) and stop.
There are no obstacles, other flowers, or people on the grid.



Start



Finish



Step 2: Analyse the problem

- Katlego is on the top left block of the grid (position (0, 0))
- The flower must be planted at position (3, 2)
- The flower is exactly three spaces ahead of Katlego.
- The flower is to be planted exactly two spaces down (South) of its current location.
- Katlego is to finish facing right (East) one space right (East) of the planted flower.
- There are no obstacles or other people to worry about.

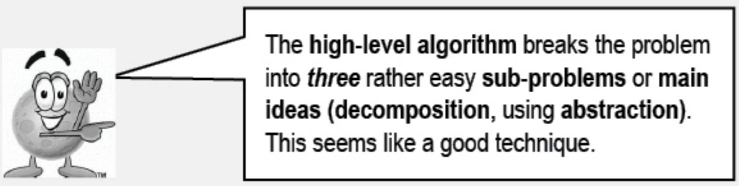
Step 3: Develop a high-level solution or algorithm (abstraction)

Katlego should do the following:

Step 1: Get the flower.

Step 2: Plant the flower.

Step 3: Move East (Forward)



Step 4: Detailed Algorithm (Decomposition)

Katlego should do the following:

1. Step 1: Get the flower.
    1.1. Move 3 blocks forward.
    1.2. Pick the flower.
2. Step 2: Plant the flower.
    2.1. Turn right.
    2.2. Move 2 two blocks forward.
    2.3. Plant the flower.
3. Step 3: Move East (Forward)
    3.1. Turn left.
    3.2. Move one block forward.

Each step in the **high-level algorithm** was broken down into more specific, detailed steps, giving more detailed instructions.

Step 5: Implement and Test the algorithm.

- Draw a grid and put two objects (one representing Katlego and one representing the flower) in the correct positions on the grid.
- Follow the algorithm and move the objects representing Katlego and the flower according to the instructions (algorithm)
- Ask the following questions:
    - Was the flower successfully moved from its first position to the target position?
    - If the answer is yes, the problem is solved else you need to identify the error and fix the algorithm (debug).

Generally, problem-based learning

- enables learners to develop problem solving strategies as well as subject knowledge and skills.
- enables learners to be more engaged in learning.
- stimulates critical thinking.
- promotes self-directed learning as learners generate problem-solving strategies.
- promotes metacognition as learners compare and reflect on solutions.
- assesses learning in ways which demonstrate understanding and competency.

See **Section 4.2** for problem-based learning assessment guidance.

PBL could incorporate strategies such as cooperative learning where learners work in small groups to solve a coding or robotics problem or use pair programming where learners work in pairs to solve coding or robotics problems.

2.7.2 Cooperative Learning

Cooperative learning is an active teaching-learning strategy where learners work in small groups, they help each other learn and in doing so, increase their joy and skills in the learning process.

Learning activities and roles are structured and overseen by the teacher, and each member of the group oversees the academic performance of the others. To successfully implement cooperative learning, leading authors in the field (David Johnson and Roger Johnson) emphasise the intentional stimulation of five basic elements (Johnson & Johnson, 2021:55-56) namely:

- **Positive interdependence**: Learners should feel like they are linked in such a way that one cannot succeed unless all in the group succeeds. Teachers should thus find ways of stimulating positive interdependence in their group activities – one possibility is giving learners different roles to fulfil; hence the group cannot move forward unless all roles are successfully fulfilled.
- **Individual accountability**: Learners should know that all will be assessed individually as well. "*The purpose of cooperative learning groups is to make each member a stronger individual in his or her right*". One way of stimulating individual accountability is by giving learners individual marks for how well they contributed to the group activity – this assessment can occur either via teacher assessment or peer assessment – by doing this, everyone will know that they cannot get a freeride during the group activity as their inputs are also individually assessed.

- **Promotive interaction**: Learners' successes are increased due to the sharing of resources, support provided, and praise and encouragement given by their group members. Teachers thus need to stimulate promotive interaction, which can be done by giving different resources to different learners. Giving learners different roles also stimulate promotive interaction.
- **Social skills**: Stimulating social skills becomes an intentional endeavour of the teacher. Teachers could provide learners with resources on how to effectively form part of a team, how to communicate well and how to resolve conflict should it arise.
- **Group processing**: Group processing forms part of reflection during and after the group activity. Teachers can stimulate group processing by giving learners a reflection sheet or by asking them open-ended questions to stimulate reflective conversations. Questions such as: "What worked well during your group activity"? or "*Describe the best experiences and worst experiences of the group activity*".

Cooperative learning can improve the learner's performance and teaches the value of teamwork, cooperation, communication, self-denial, and initiative taking.

2.7.2.1 Implementing cooperative learning in Foundation Phase Coding and Robotics

Example of cooperative learning activity for foundation phase on the topic of robotics (Table 3, Grade 2):

"Learners present the concept that a robot comprises of different components, each with a purpose. Reference is made to moving parts, sensors"

The group's task is to use the flashcards provided (graphically illustrating what a robot is, examples of robots and moving/sensory parts of a robot) and to draw their own robot.

Divide the class into groups of four (4) learners. Each learner gets a role of a robot's moving and/or sensory parts:

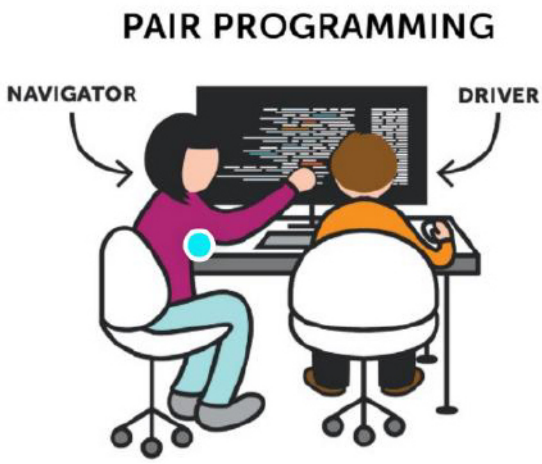
- **Learner 1 (Arms) –** Learner that is responsible to get all the resources together (e.g., flashcards of what a robot is, examples of robots etc.).
- **Learner 2 (Light sensor) –** Learner that is responsible to ask "Why". As foundation phase learners are naturally inquisitive, having a learner responsible to keep asking "Why" would lead the group to critical thinking.
- **Learner 3 (Wheel) –** Learner that draws the robot and follows instructions from the other group members.
- **Learner 4 (Sound sensor) –** Learner that presents the group's robot drawing and explains what they think a robot is and what the different parts of their robot are.

Refer to Annexure A for cooperative learning assessment guidance.

Pair programming could also be used as a cooperative teaching and learning strategy to solve programming problems.

2.7.3 Pair Programming

Pair programming is a pedagogical approach that involves two learners working together on one computer or one piece of paper to complete a shared goal/task. It emanates from the programming industry yet has proven to be successful even at school level. One of the learners fulfils the role of the "*driver*" and one of the learners fulfils the role of the "*navigator*".



The driver is the learner who may use the computer and handle the keyboard, or draw on the paper and handle the pencil, whereas the navigator is the learner who utilises the resources, and reviews the driver's work throughout, provides feedback and suggestions to the driver, points out errors and asks questions to the teacher. Pair programming is a collaborative effort that involves a lot of communication, discussion, and problem-solving.

Although pair programming can be implemented as a collaborative "unstructured" pair activity, it is best to stimulate the five basic elements of cooperative learning, as described above when implementing pair programming in the classroom.

It also appears particularly promising in situations where there are not enough computing devices for learners to work individually as well as for increasing learning and engagement with technology among learners with limited device experience. It is also suggested that learners show higher confidence when programming in pairs. It allows learners to share knowledge and learn from each other, improves learning engagement, and teach each other.

2.7.3.1 Implementing pair programming in Foundation Phase Coding and Robotics

Example of pair programming activity for foundation phase on the topic of Coding (see Table 2 Grade 1):

"*A given pattern is identified. A given pattern is extended. A simple pattern is created by the learner and repeated*".

The pair's task is to identify the pattern from the resource given, fill in the blanks by repeating the pattern and then draw their own pattern with the same sequence.

- **Driver –** The learner acting as the driver will be the one completing the pattern and drawing the pattern decided upon between the two learners.
- **Navigator –** The learner acting as the navigator may have flashcards with similar patterns (different pictures) on them. They may also ask the teacher for help.

Note:
The teacher may swop the learners' roles as the activity progresses to ensure that both learners have a chance to fulfil each role. You may also ask any one of the learners to present their work to the class. This ensures that both learners feel a need to engage and gives more learners an opportunity to practice communication skills.

Refer to Annexure A for pair programming assessment guidance.

2.7.4 Deliberate Practise

A subject such as coding and robotics not only requires thinking skills, but also requires focused teaching and ample practise. This practise should, however, be purposeful, well thought through with gradual increase in complexity.

The curriculum is designed to encourage deliberate practise, as competencies are repeated within and across grades. The concept of deliberate practise is particularly focused on skill acquisition and development and is key in the development of competency and expertise in subjects such as coding.

Deliberate practise is a specific type of practise that involves setting specific goals, receiving feedback, and making focused efforts to acquire and improve skills and performance. It is not simply repeating skills over-and-over again, but rather adjusting to improve competencies as well as gradually adding additional competencies that lead to mastery. It therefore involves purposeful repetition, feedback-driven metacognition, and extension to improve performance (Ericsson, 2008; Deans for Impact, 2016; Ericsson *et. al.*, 2018).

In terms of extension, deliberate practice involves extending the amount of time spent practising, adding new features, and increasing the complexity of tasks. The goal is to push beyond one's comfort zone to achieve growth and improvement.

2.7.5 Science of Learning

The curriculum is also informed by the Science of Learning, a multidisciplinary field that combines research from cognitive psychology, neuroscience, educational psychology, and other related disciplines to understand how people learn. It also aims to identify the most effective teaching and learning strategies based on empirical evidence that has been shown to improve long-term retention of information and enhance learning outcomes.

Learning is an iterative process that requires that one continually revisits what one has learned earlier, update it, and connect it with new knowledge. Learning always builds on a store of prior knowledge and is the residue of thought. New learning requires a considerable amount of practise and meaningful connections to existing knowledge. Learning, therefore, requires learners thinking (Brown *et al*., 2014; Dereck Bok Center, Harvard University, 2023).

Science of learning includes the following learning strategies (*Weinstein et* al., 2018):

- **Retrieval practice**: Bringing learned information to mind from long-term memory.
- **Spaced practice**: Spreading learning activities out over time/reviewing previously learned information at gradually increasing intervals.
- **Interleaving**: Switching between topics while learning.
- **Examples**: When learning abstract concepts, illustrating them with various examples or concrete experiences.
- **Dual coding**: Combining words with visuals.
- **Elaboration**: Classroom discussions that require learners to relate new material to what they already know and to recall previously learned information, including asking *why* and *how* questions with learners explaining in their own words.
- **Interactive activities**: Engage actively with learning material using activities that require one to retrieve (recall) previously learned information.

2.8 SYNERGISING CODING AND ROBOTICS IN FOUNDATION PHASE

In the foundation phase, elements of subjects such as Language, Mathematics and Life Skills can be linked to Coding and Robotics and therefore be integrated into coding and robotics to enhance the learning experience. For example:

Algorithms involve sequencing and summarising in literacy and breaking down complex problems into simpler steps in mathematics.

Modularity: Involves breaking down tasks into manageable units in computer science, while in mathematics, it involves breaking down a complex problem into smaller, manageable parts.

Control structures in coding determine how a set of instructions are executed within a program, while heuristic thinking in mathematics involves using logical thinking and trial and error to solve problems.

Coding and natural language: The process of learning to code is also often likened to language acquisition, as learners progress through six distinct stages of understanding. These stages bear close resemblance to the stages of literacy development.

Design: Designing robotics artefacts links to aspects of Creative Arts as part of Life Skills.

Digital concepts: Aspects such as the impact of technology and being a digital citizen links to Life Skills (Personal and Social well-being).

By developing these skills in Coding and Robotics, learners can develop habits of mind that will be valuable in other subjects.

2.9 TIME ALLOCATION

In Grades R, 1 and 2, 1 hour per week (10 hours per term) is allocated for coding and robotics. In Grade 3, the time allocation for coding and robotics is 2 hours per week (20 hours per term).

The following table provides the time allocation per term as a percentage of the total available time per term:

Table 2.1: Time allocation for Foundation Phase Coding and Robotics

Gr R ,1,2 = % per week Gr 3 = % per week	Term 1				Term 2				Term 3				Term 4			
	R	1	2	3	R	1	2	3	R	1	2	3	R	1	2	3
Pattern Recognition	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
Algorithms & Coding	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
Robotics	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
Digital Concepts	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
Total	10 weeks				10 weeks				10 weeks				10 weeks			

Note:

Sections 2.12.1 (coding content) 2.12.2 (robotics content), 2.12.3 (digital concepts content) and Section 3 (unpacking of the content) describe many concepts and competencies across the three strands that are linked and support each other. Various competencies across the three strands can therefore be linked and dealt with in an integrated fashion.

2.10 Resources Required to Offer Coding and Robotics in Foundation Phase

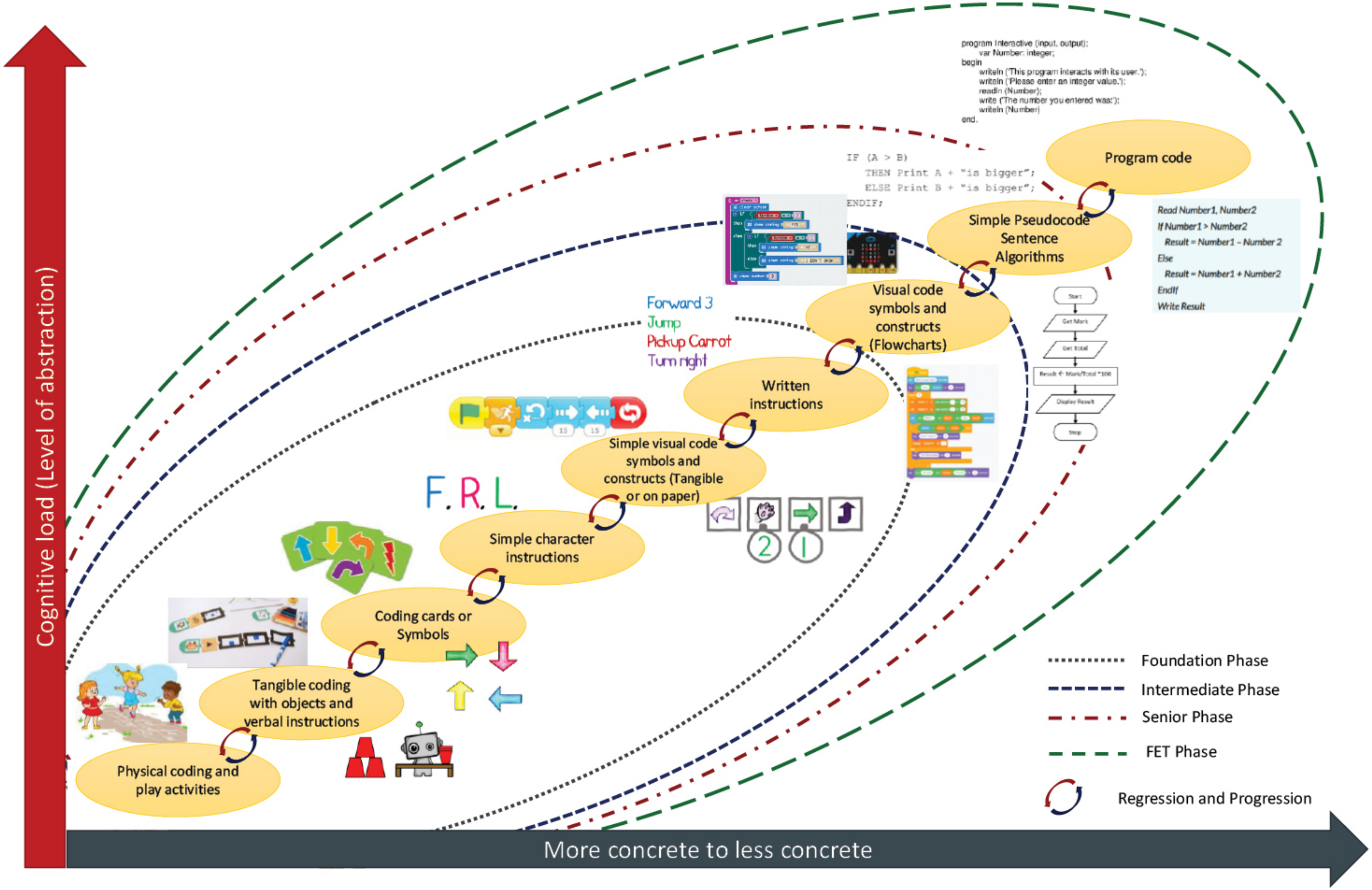


Figure 2.9 Programming resources for Coding and Robotics Coding Resources

Refer to Figure 2.9:

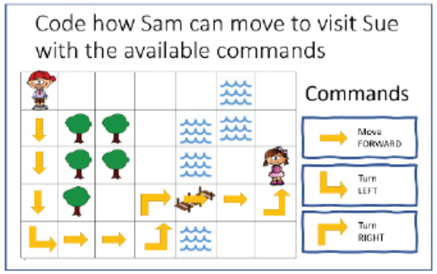
> "Computer programming is a highly cognitive skill, which requires mastery of multiple domains, and is acknowledged as being difficult to learn, making it essential to take into account the cognitive loads (CLs) imposed on learners, as well as their abilities to absorb this knowledge during the teaching and learning process". *Berssanette & de Francisco (2022).*

Foundation phase follows an unplugged programming approach. Literature suggests that interactive unplugged programming in early years of education significantly influence improving learners' performance of computational thinking skills and learning engagement and it also confirms the teaching value of interactive unplugged programming (Li *et al.*, 2023). An unplugged approach therefore reduces cognitive loads as it helps learners to the learn the foundational concepts and principals of computational thinking and coding without getting overwhelmed by the intricacies of programming environments. It therefore serves as an effective steppingstone for beginners to develop their problem-solving and programming skills before transitioning to coding environments.

Unplugged codding resources include:

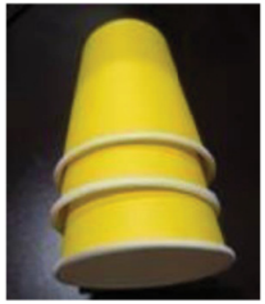
- Outside grid on play area or in class
- Coding cards (arrows or symbols)
- Playful artefacts/toys used in coding problems (e.g., Flowers, sweets, insects) – Bee collecting nectar, Robot sorting trash (trash items, cans, plastic bottles) – Paper cups – Egg holders – Pom pom's – Ice cream sticks – Simple uni-fix blocks.
- *Optional: Free and open-source educational software/apps, e.g., Scratch Jnr with a device such as a tablet or PC
- *Optional: programmable educational robots
- Learners acting as robots with simple props (Box) etc.

Outside Grid



Coding activity sheets and coding cards

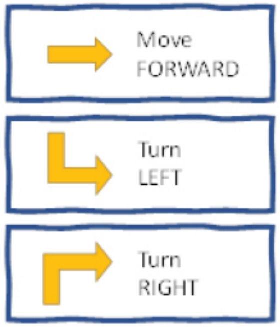
Strings and beads

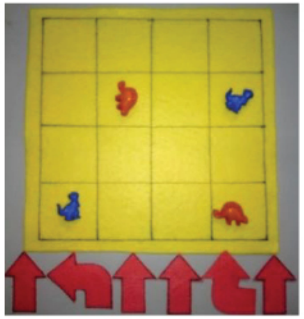


Paper cups

*Note: Software and/or devices are optional as the curriculum is designed to be implemented without devices. However educational devices, where available, could be used for enrichment.

Coding Cards





Felt grid and arrrows with plastic buttons

Pipe cleaners, pom poms, recycled bottle caps, ice cream sticks.

Beads

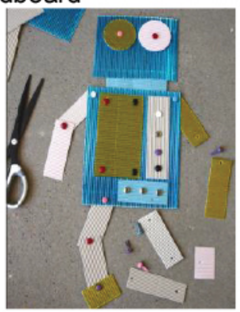
2.10.1 Robotics Resources

Refer to Figure 2.9:

- Basic stationary (cardboard, coloured paper. blocks books, rubber bands, glue)
- Other material such as uni-fix/counting blocks, pom-pom's, string, pipe cleaners, marbles.
- Recycled items (newspaper, brown paper bags, used toilet rolls, used boxes (e.g., cereal or pill boxes/holders, egg holders, trash items such as used cooldrink cans, plastic bottles, bottle caps), paper cups, straws.
- Salted playdough (wires, battery, light bulb)
- *Optional: Programmable educational robots/virtual robots [Scratch Junior → object, e.g., CAT, is an example of a virtual robot]
- Pictures of different types of robots and robot components
- DC motor, battery pack, small light bulb

Outside Grid

Split pins/paper fasteners with cardboard



Twine, straws, glue, paper binders, rubber bands

DC motor, battery pack, light bulb and holder (optional). Clay/Play dough. Split pins

Different types of old cardboard boxes

Stones and Marbles

2.10.2 Digital Concepts Resources

- Sample technologies and components (e.g., Mobile phone, tablet, Laptop (with screen, keyboard and mouse), etc.)
- Pictures of computing devices, input devices, output devices

2.11 Overview of Foundation Phase Coding and Robotics

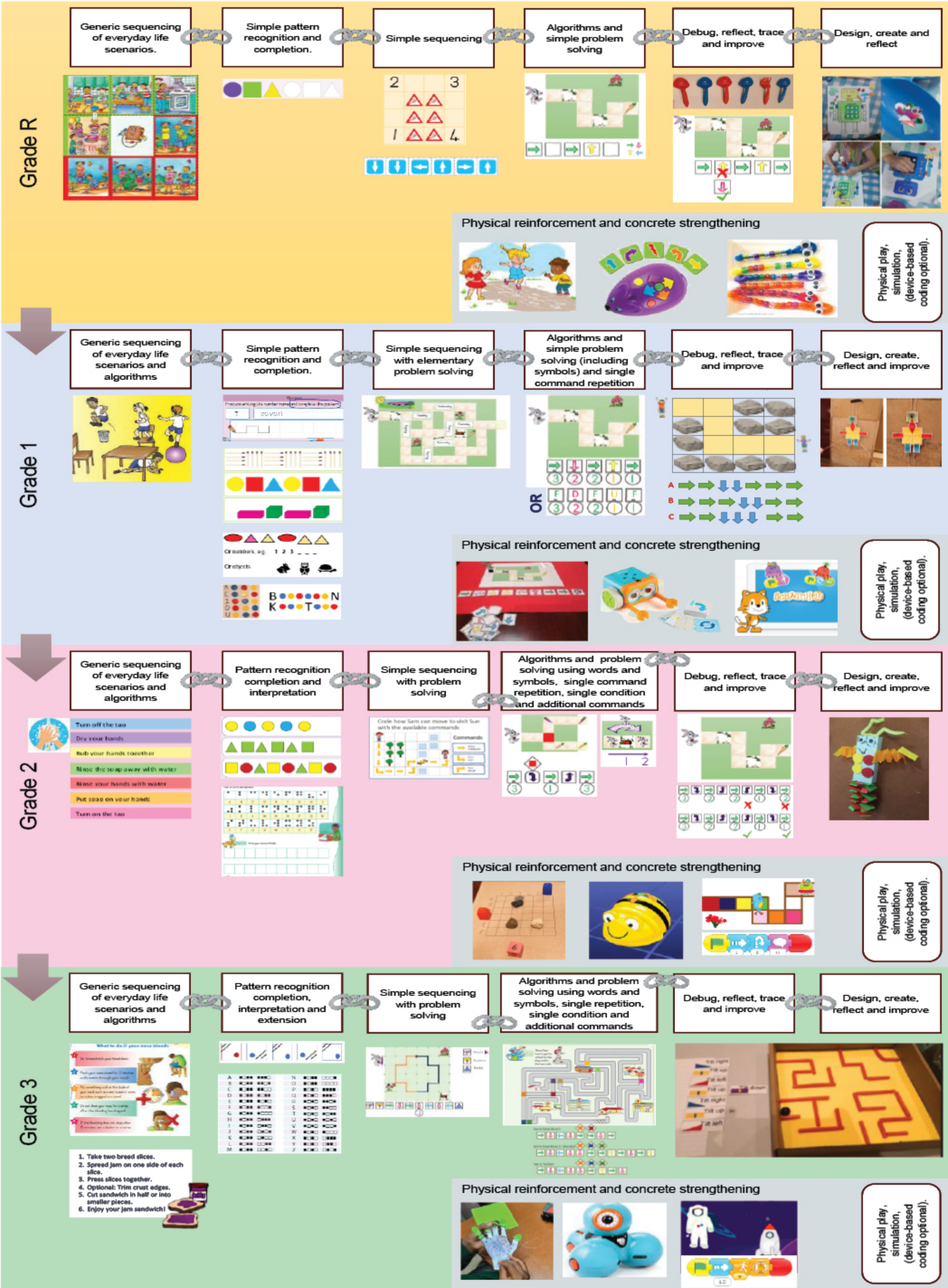


Figure 2.10 Overview of Foundation Phase Coding and Robotics

2.12 FOCUS OF CONTENT AREAS

2.12.1 Coding

Table 2.2: Coding content focus

Competency	Grade R	Grade 1	Grade 2	Grade 3
	(Novice level)			
C.1 **Apply computational thinking skills to develop a set of logical instructions to solve a problem.**	Rudimentary operations are presented using sequences of pictures. Logically order a set of pictures to accomplish a set task. Order, arrange or search a set of pictures and symbols according to given criteria.	Rudimentary operations are presented using sequences of pictures and or elementary three-word sentences. Logically order a set of pictures or three-word sentences to accomplish a set task. Order, arrange or search a set of pictures, symbols, characters, and numbers according to given criteria.	Elementary tasks and logical instructions are identified to solve a problem. Elementary operations are presented using sequences of pictures and or simple sentences. Logically order a set of pictures or simple sentences to accomplish a set task. Order, arrange or search a set of pictures, symbols, characters, numbers, and words according to given criteria.	Foundational tasks and logical instructions are identified to solve a problem from which unnecessary or irrelevant details are ignored. Foundational operations are presented using sequences of pictures and or simple sentences. Logically order a set of pictures, simple sentences to accomplish a set task. Order, arrange or search a set of pictures, symbols, characters, numbers, and words or sentences according to given criteria.
C.2 **Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.**	Symbols are used to represent actions and operations to accomplish a task. Each symbol represents a single task. The solution may be presented partially requiring the learners to complete a problem. Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment scenarios	Symbols are used to represent actions and operations to accomplish a task. Symbols may be grouped to represent repetition. The solution may be presented partially, requiring the learners to complete it. Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment scenarios	Symbols or written statements are used to represent actions and operations to accomplish a task. Symbols may be grouped to represent repetition. Symbols / blocks may be used to represent a condition. Symbols may include block-code type images with linkages. The solution may be presented partially, requiring the learners to complete it. Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment scenarios	Symbols (normal or puzzle type or written statements are used to represent actions and operations to accomplish a task. Symbols / puzzles blocks may be grouped to represent repetition (or a statement indicating repetition) Symbols/ blocks may be used to represent a condition (or a statement indicating condition) Symbols may include block-code images with linkages. The solution may be presented partially, requiring the learners to complete it. Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment scenarios
C.3 **Interpret and execute a given symbolic or written set of commands**	A rudimentary set of commands in relation to C.2. are correctly executed physically, on paper or with an educational tool. One learner could take on the role of instructor and or interpreter(executer)	A rudimentary set of commands in relation to C.2 are correctly executed physically, on paper or with an educational tool. One learner could take on the role of instructor and or interpreter(executer)	An elementary set of commands in relation to C.2 are correctly executed physically, on paper or with an educational tool. One learner could take on the role of instructor and or interpreter(executer)	A foundational set of commands in relation to C.2 are correctly executed physically, on paper or with an educational tool. One learner could take on the role of instructor and or interpreter(executer)
C.4	A rudimentary set of	A rudimentary set of	An elementary set of	A foundational set of

Debug a given symbolic or written set of instructions.	commands to solve a problem is inspected for an error and corrected. Debugging relates to a code set or set of instructions in relation to C.1., C.2. and C.3.	commands to solve a problem is inspected for an error and corrected. Debugging relates to a code set or set of instructions in relation to C.1., C.2. and C.3.	commands to solve a problem is inspected for an error or errors and corrected. Debugging relates to a code set or set of instructions in relation to C.1., C.2 and C.3.	commands to solve a problem is inspected for an error or errors and corrected. Debugging relates to a code set or set of instructions in relation to C.1., C.2. and C.3.
C.5 **Evaluate a given solution towards potential improvement.**	Reflect and report on a given solution. Ask the following questions (critical thinking): • What happened? • Why has it happened?	Reflect and report on a given solution. Ask the following questions (critical thinking): • What happened? • Why has it happened? • What can be learnt? The learners are given the opportunity to reflect on their thinking. A rudimentary set of commands to solve a problem is inspected and an alternate is suggested. The evaluation relates to a code set or set of instructions in relation to C.1, C.2 and C.3	Reflect and report on a given solution. Ask the following questions (critical thinking): • What happened? • Why has it happened? • What can be learnt? The learners are given the opportunity to reflect on their thinking. An elementary set of commands to solve a problem is inspected and an alternate is suggested. Incorporating: • Reducing the numbers of steps The evaluation relates to a code set or set of instructions in relation to C.1, C.2, C.3	Reflect and report on a given solution. Ask the following questions (critical thinking): • What happened? • Why has it happened? • What can be learnt? The learners are given the opportunity to reflect on their thinking. A foundational set of commands to solve a problem is inspected and a better alternate is suggested. Incorporating: • Reducing the numbers of steps • Grouping of repetitive steps The evaluation relates to a code set or set of instructions in relation to C.1, C.2, C.3
C.6 **Recognise and interpret patterns in symbolic sets of data or visualisations.**	A rudimentary pattern is identified incorporating a single set of elementary: numbers, pictures, colours, symbols, or shapes. The differences and or similarities between sets of data patterns including images are identified and motivated. The learners can recognise and explain the composition of the pattern. The pattern is copied by the learner either using physical objects, physical (kinaesthetic) movements or on paper.	A rudimentary pattern is identified incorporating a single set of elementary: numbers, pictures, colours, symbols, or shapes. The differences and or similarities between sets of data patterns including images are identified and motivated. The learners can recognise and explain the composition of the pattern. The pattern is copied by the learner either using physical objects, physical (kinaesthetic) movements or on paper.	An elementary pattern (which could include an inverse) is identified incorporating a single set of elementary: numbers, pictures, colours, symbols, or shapes. The differences and or similarities between sets of data patterns including images are identified and motivated. The learners can recognise and explain the composition of the pattern. The pattern is copied by the learner either using physical objects, physical (kinaesthetic) movements or on paper.	A foundational pattern (which could include an inverse or a grid artefact) is identified incorporating a single set of elementary: numbers, pictures, colours, symbols, or shapes. The differences and or similarities between sets of data patterns including images are identified and motivated. The learners can recognise and explain the composition of the pattern. The pattern is copied by the learner either using physical objects, physical (kinaesthetic) movements or on paper.
C.7 **Create or complete a pattern to represent a data set.**	N/A	Identify a given pattern. Extend a given pattern. Create and repeat a rudimentary pattern. Pattern conforms to the prescriptions of C.6.	Identify a given pattern. Extend a given pattern. Create and repeat an elementary pattern and explain its composition. Pattern conforms to the prescriptions of C.6.	Identify a given pattern. Extend a given pattern. Create and repeat a foundational pattern and explain its composition. Pattern conforms to the prescriptions of C.6.

2.12.2 Robotics

Table 2.3: Robotics content focus

Competency	Grade R	Grade 1	Grade 2	Grade 3
	(Novice level)			
R.1 **Explain what a robot is in simple terms.**	Present a rudimentary explanation of what a robot is.	Present an elementary explanation of what a robot is, including reference their purpose.	Present an elementary explanation of what a robot is, including reference to their purpose and mode of operation. Reference to moving and sensory parts are made.	Present a foundational explanation of what a robot is, including reference to their purpose and mode of operation. Reference to moving, sensory and processing parts are made.
R.2 **Identify different types of robots.**	Identify general examples of robots.	Identify general examples of robots and what they do.	Identify robots that are used in factories and robots that are not used in factories (Service robots)	Identify domestic robots and professional use robots.
R.3 **Outline the different components of a robot**	N/A	The learners present the concept that a robot comprises of different components each with a purpose. Reference is made to moving parts, sensors.	The learners present the concept that a robot comprises of different components, each with a purpose. Reference is made to sensors, a power source and motors	The learners present the concept that a robot comprises of different components, each with a purpose. Reference is made to the following concepts as part of the outline: • Robots comprise of mechanical parts. • Requires power. • Require some form of programming.
R.4 **Present an understanding of how robots affect the world.**	Provide a rudimentary explanation of what robots are used for.	Provide a rudimentary explanation of what robots are used for with references to specific tasks.	Provide an elementary explanation of what robots are used for with references to specific tasks, including dangerous and repetitive ones.	Provide a foundational explanation of what robots are used for with references to specific concepts that robots can be programmed to react to their environment. The discussion incorporates elements of R.1 and R.2
R.5 **Design a simple artefact based on a set of design specifications.**	Create a rudimentary artefact to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk / ideate about what their robots can do.	Create a rudimentary artefact to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk / ideate about what their robots can do. Strings and /or pins may be added to mimic movement. Different materials can be used, e.g., pipe cleaners, ice cream sticks, straws etc. The creation of the artefact could also take on the form of a game e.g. (Assemble by	Create an elementary artefact to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk / ideate about what their robots can do. Strings and/ or pins may be added to mimic movement. Different materials can be used, e.g., pipe cleaners, ice cream sticks, straws etc. The creation of the artefact could also take on the form of a game e.g. (Assemble by	Create a foundational artefact to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk / ideate about what their robots can do including the composition of the various parts and the purpose of each. Strings and/ or pins or lever mechanisms and /or pullies ***may*** be added to mimic movement. Different materials can be

		numbers)	numbers), throw some die. Assemble using prefabricated parts if (available) e.g., building blocks. The instructions contain various steps that should be read and or interpreted as part of the assembly. The assembly should require a set order (one step should follow the other)	used, e.g., pipe cleaners, ice cream sticks, straws etc. The creation of the artefact could also take on the form of a game e.g. (Assemble by numbers), throw some die. Assembly using prefabricated parts if (available) e.g., building blocks. The instructions contain various steps that should be read and/or interpreted as part of the assembly. The assembly should require a set order (one step should follow the other)
R.6 **Mimic the operations of a robot**	The learners mimic the operations of a robot based on given instruction or for a purpose. Rudimentary instructions are performed, in person or using a tool, or on paper. Relates to C.1, C.2, C.3, C.4 and C.5.	The learners mimic the operations of a robot based on given instruction or for a purpose. Rudimentary instructions are performed, in person or using a tool or on paper. Relates to C.1, C.2, C.3, C.4 and C.5.	The learners mimic the operations of a robot based on given instruction or for a purpose. Elementary instructions are performed, in person or using a tool, or on paper. Relates to C.1, C.2, C.3, C.4 and C.5.	The learners mimic the operations of a robot based on given instruction or for a purpose. Foundational instructions are performed, in person or using a tool, or on paper. Relates to C.1, C.2, C.3, C.4 and C.5.
R.7 **Create, test, and execute a set of robotic instructions.**	A rudimentary set of instructions are compiled and executed to perform a task. This outcome and instructions relate to C.1, C.2, C.3, C.4 and C.5. The execution can also be done and demonstrated in the context of R.6	A rudimentary set of instructions are compiled and executed to perform a task. This outcome and instructions relate to C.1, C.2, C.3, C.4 and C.5. The execution can also be done and demonstrated in the context of R.6	An elementary set of instructions are compiled and executed to perform a task. This outcome and instructions relate to C.1, C.2, C.3, C.4 and C.5. The execution can also be done and demonstrated in the context of R.6	A foundational set of instructions are compiled and executed to perform a task. This outcome and instructions relate to C.1, C.2, C.3, C.4 and C.5. The execution can also be done and demonstrated in the context of R.6

2.12.3 Digital Concepts

Table 2.4: Digital Concepts content focus

Competency	Grade R	Grade 1	Grade 2	Grade 3
	Novice Level			
D.1 **Outline the concept of technology and purpose of information technology (IT).**	Present a rudimentary explanation of what technology is. Learners can point out examples of technology.	Present a rudimentary explanation of what technology is. Learners can point out examples of technology and relate its use to everyday life. Learners relate the concept of technology to that of a tool.	Present an elementary explanation of what technology is. Learners can point out examples of technology and relate its use and purpose to everyday life. Learners relate the concept of technology to that of a tool.	Present a foundational explanation of what technology is. Learners can point out examples of technology and relate its use and purpose to everyday life. Learners relate the concept of technology to that of a tool. The learner's answer include that the technological artefact has a common purpose or goal. The answer also includes the concept that technologies often comprise of different components.
D.2 **Recognise that he or she is living as citizens in a digital world.**	The learners present a rudimentary understanding that the digital world is all around us. The learners understand that electronic devices (dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and in moderation (screen time)	The learners present a rudimentary understanding that the digital world is all around us. The learners understand that electronic devices (dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and in moderation (screen time) Present a rudimentary understanding of the dangers of going online. Present a rudimentary understanding of the concept of cyberbullying and how to deal with it. The conceptualisation is presented in terms of D.1	The learners present an elementary understanding that the digital world is all around us. The learners understand that electronic devices (dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and in moderation (screen time) Present an elementary understanding of the dangers of going online. Present an elementary understanding of the concept of cyberbullying and how to deal with it. The learners understand that protecting information with a password helps keep it private. Introduce the concept of a digital footprint at an elementary level. The conceptualisation is presented in terms of D.1	The learners present a foundational understanding that the digital world is all around us. The learners understand that electronic devices (dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and in moderation (screen time) Present a foundational understanding of the dangers of going online. Present a foundational understanding of the concept of cyberbullying and how to deal with it. The learners understand that protecting information with a password helps keep it private. Recognise the concept and dangers of sharing personal information like usernames and or passwords. Understand the responsible use of technology. Introduce the concept of a digital footprint at a foundational level. Present an understanding of the necessity to report unsuitable use of electronic communication, the access of inappropriate content and or contact.

				The conceptualisation is presented in terms of D.1
D.3 **Demonstrate an understanding of the concept of a computing device.**	The learner presents a rudimentary explanation of what a computing device is. Learners can point out examples of computing devices.	The learner presents a rudimentary explanation of what a computing device is. Learners can point out examples of computing devices. The learner's answer should incorporate the concept that a computing device can follow and interpret instructions. Links with D.1	The learner presents an elementary explanation of what a computing device is. Learners can point out examples of computing devices. The learners answer should incorporate the concept that a computing device can follow and interpret instructions. Links with D.1	A foundational explanation of what a computing device is, is presented. Learners can point out examples of computing devices. The learners answer should incorporate the concept that a computing device can follow and interpret instructions and produce output/result or render an outcome. Links with D.1
D.4 **Identify the common uses of ICT in the real world.**	N/A	A rudimentary list of the use of IT related technologies and devices are named in terms of their use. Links with D.1 and D.2	An elementary list of the use of IT related technologies and devices are named in terms of their use. Links with D.2 and D.2	A foundational list of the use of IT related technologies and devices are named in terms of their use. Links with D.1 and D.2
D.5 **Differentiate between the components of an ICT system.**	N/A	N/A	The learners differentiate at a rudimentary level between the concept of hardware (touchable) and software as "Apps". Basic examples are listed in relation to D.1, D.2 and D.3	The learners differentiate at an elementary level between the concept of hardware (touchable) and software as "Apps". Learners lists examples of different types of hardware and software. Basic examples are listed in relation to D.1, D.2 and D.3
D.6 **Explain how the adaptation of technology impacted the world we work and live in.**	N/A	N/A	The learners can present a rudimentary explanation of how technology impact society at large. The discussion incorporates concepts of D.1 and D.2	The learners can present an elementary explanation of how technology impact society at large. Typical examples are listed in relation to various sectors. The discussion incorporates concepts of D.1 and D.2
D.7 **Present a basic understanding of the concept of input processing and output.**	The learners present a rudimentary understanding that input results in some form of output.	The learners present a rudimentary understanding that input results in some form of output. Input → Instructions are executed that results in an action. The concept that different forms of input results in different actions are emphasised.	The learners present an elementary understanding that input results in some form of output. Input → Instructions are executed that results in an action. Output as a form of communication from the device The concept that different forms of input results in different actions is emphasised.	The learners present foundational understanding that input results in some form of output. Input → Instructions are executed that results in an action. Output as a form of communication from the device The concept that different forms of input result in different actions is emphasised. The concept that processing takes place between input and output forms part of the learners understanding.
D.8 **Interpret a pattern to represent or**	Interpret a rudimentary pattern and present a corresponding message in	Interpret a rudimentary pattern and present a corresponding message in	Interpret an elementary pattern and present a corresponding message in	Interpret a foundational pattern and present a corresponding message in

communicate a message or image.	symbolic form. Done in relation to C.6.	symbolic form. Translate (decode) an elementary pattern to a simple word, image, or phrase (3-word maximum phrase). Done in relation to C.6	symbolic form. Translate (decode) an elementary pattern to a simple word, image, or basic sentence. Done in relation to C.6	symbolic form. Translate (decode) a foundational pattern to a simple word, image, or simple sentence. Pattern may include 2-D matrixes for both encoding and decoding purposes. Done in relation to C.6
D.9 **Create a pattern to represent or communicate a message or image.**	Create a rudimentary pattern to represent an image or communicate a message or an image. Done in relation to C.6 and D.8.	Create a rudimentary pattern to represent an image or communicate a message or an image. Translate (encode) a rudimentary pattern to a simple word, image, or phrase (3-word maximum phrase). Done in relation to C.6 and D.8.	Create an elementary pattern to represent an image or communicate a message or an image. Translate (encode) an elementary pattern to a simple word, image, or basic sentence. Done in relation to C.6.and D.8	Create a foundational pattern to represent an image or communicate a message or an image. Translate (encode) a foundational pattern to a simple word, image, or simple sentence. Done in relation to C.6 and D.8.
D.10 **Demonstrate a basic proficiency in the application of digital skills.**	**N/A**	**N/A**	**N/A**	**N/A**

2.13 Envisaged Learner

The Coding and Robotics learner shows an interest in technology and its application in the world. The learner can think logically and critically and is able to solve problems. Furthermore, the learner is creative and innovative as well as disciplined, focused, and persistent. The learner can also work well with others to achieve a common goal.

2.14 Career Opportunities

Today, digital technologies are integrated in all aspects of our lives. Digital competencies such as Coding and Robotic skills make one more employable and effective in any job and support further studies.

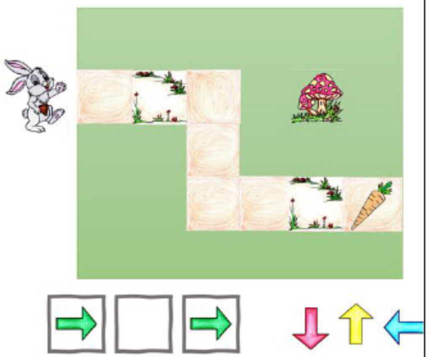
The growing ubiquity of digital technologies and the developments around the Internet of Things (IoT), automation and artificial intelligence (AI) have seen the inclusion of skills such a computational thinking, design thinking, software development (coding) and robotics in every sector of employment and entrepreneurship. Therefore, Coding and Robotics aims to equip learners with knowledge and skills that will allow them to thrive in any career and specifically in careers such as software development, robotics engineering, artificial intelligence, etc.

2.15 Progression and Exit Skills Per Grade of Focus Areas

As Coding and Robotics is a process driven subject, the exit skills cannot be broken up into terms. The exit skills must be repeated and practised in every term, but in a progressive way. Conceptual knowledge must be continuously developed to enable procedural knowledge. Some skills can integrate into Language, Mathematics and Life Skills but there are subject specific skills that can only be developed in Coding and Robotics as a separate subject.

Grade R

Supply the missing instruction for Sipho Super Bunny to follow the path and get to the carrot.

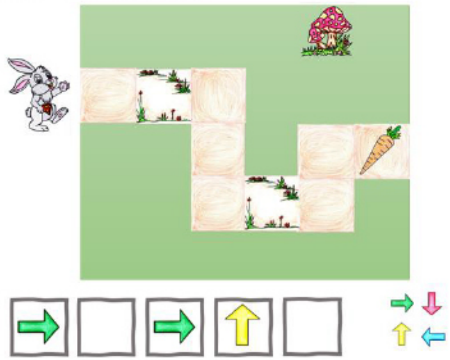


Note:
Complexity of activities increase, from Grade R to Grade 3, for example, in terms of number of instructions, more complex grids, adding restrictions or limitations. adding new instructions and adding the concept of the repetition construct.

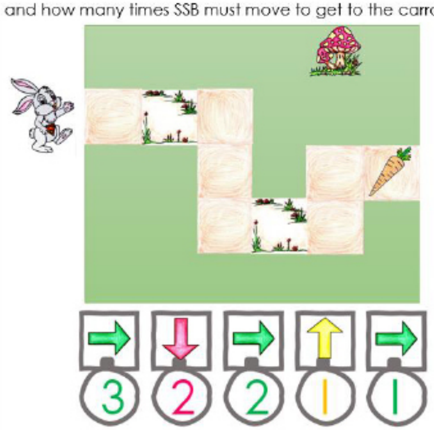
In Grades R and 1, focus on concrete, practical activities, e.g., using a grid on the floor.

Grade 1

Supply the missing instructions for Sipho Super Bunny (SSB) to follow the path and get to the carrot.

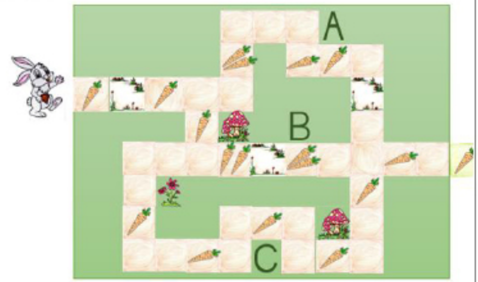


Sipho can only move one block at a time. Write down the instructions and how many times SSB must move to get to the carrot.



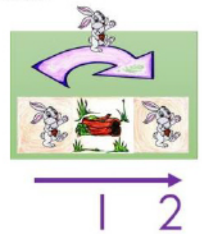
Grade 2

Study the following image and answer the questions that follow.



- Which path (A, B or C), counting the number of tiles is the longest?
- Taking which path (A, B or C) will allow SSB to collect the most carrots?

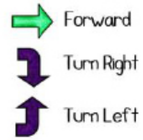
SSB can jump over logs. When he jumps, he moves 2 spaces (tiles) forward.



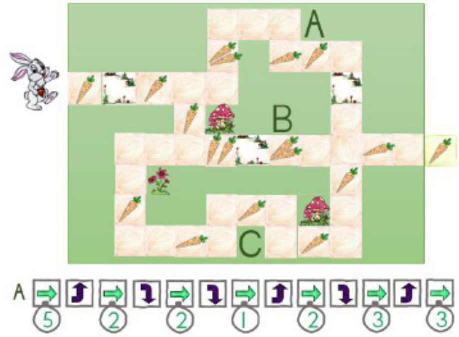
Sipho can only move one block at a time and jump Write down the instructions and how many times SSB must jump or move to get to the carrot.

Grade 3

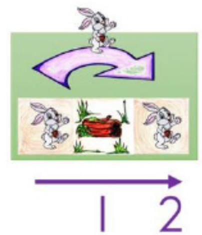
SSB can only perform the following commands.



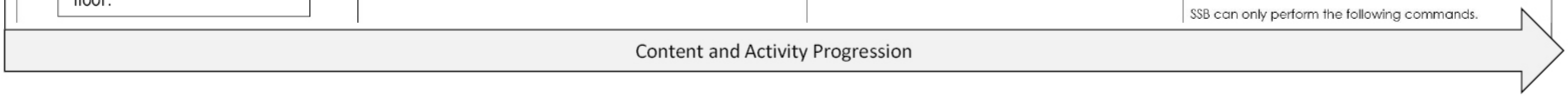
Sipho can only move one block at a time. Write down the instructions and how many times SSB must move and turn right or left to get to the green tile following path A.



SSB can jump over logs. When he jumps, he moves 2 space (tiles) forward.



SSB can only perform the following commands.



Grade R

Grade 1

Study the following image and answer the questions that follow.

- How many carrots will SSB have collected before he reaches the mushroom?
- How many carrots will SSB have collected once he has walked the whole path?

Study the following image and answer the questions that follow.

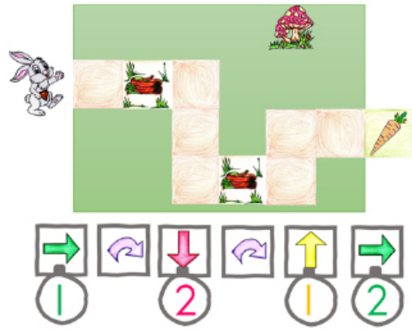
- How may carrots will SSB have collected if he stands on block A?
- How may carrots will SSB have collected if he stands on block B?
- How may carrots will SSB have once he as collected all the carrots?
- The carrots follow a certain pattern how many carrots must be placed on block C to complete the pattern?

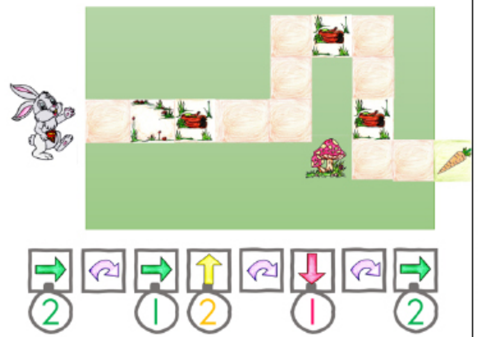
Sequencing & Problem solving

In the DBE Math English grade 1 workbook (p 69) the days of the week are given in sequence. Such an exercise can easily be converted to a sequencing and problem-solving activity. The example below can be phrased as a problem as follows:

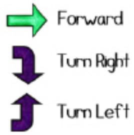
SSB must identify the days of the week in the correct order. How should SSB walk to cross each day but not cross the same tile twice?

Grade 2



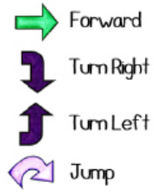


SSB can only perform the following commands.

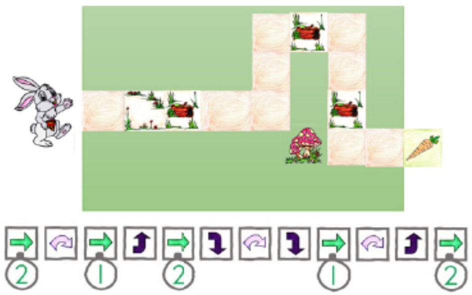


Sipho Super Bunny can only move one block at a time. Write down the instructions and how many times SSB must move and turn right or left to get to the carrot.

Grade 3



Sipho can only move one block at a time, jump and turn left and right. Write down the instructions and how many times SSB must move or jump and turn right or left to get to the green tile and carrot.



SSB can only perform the following commands.

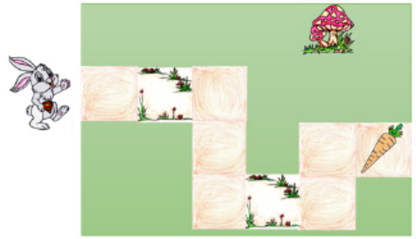
Sipho can only move one block at a time, jump and turn left and right and pick up c carrot at his existing position. Write down the instructions and how many times SSB must move or jump and turn right or left to get to the green tile and carrot.

Content and Activity Progression

Grade R

Grade 1

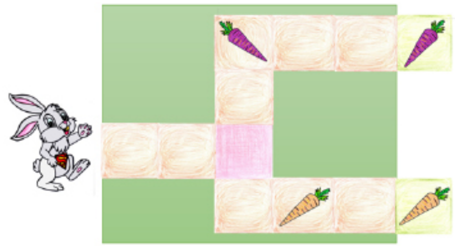
Grade 2

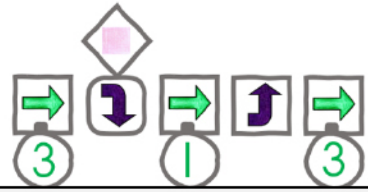


The following symbol is used to represent and If … then construct.

SSB does not like purple carrots at all. The following symbolic algorithm represents the solution on the direction that Sipho SB should turn if he lands on the pink tile. It could be interpreted as: If I am on the pink tile turn right.

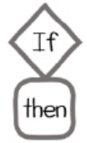
Typically, the learners will be required to supply the missing code element or to construct the solution algorithm for themselves.



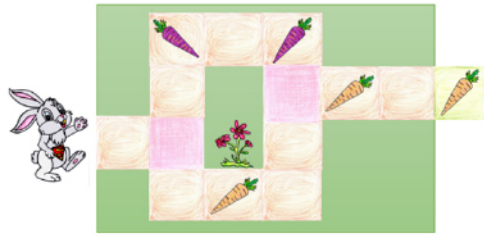


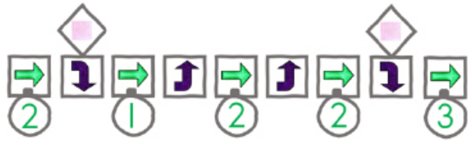
Grade 3

The following symbol is used to represent and If … then construct.



SSB does not like purple carrots at all. The following symbolic algorithm represents the solution on the direction that Sipho should turn if he lands on the pink tile. It could be interpreted as: If I am on the pink tile turn right.





Content and Activity Progression

Note:
In terms of coding, typically, problems could require learners to

- read code and explain what it does
- work through (trace) / act out code (physically or simulated) to determine the output or the correctness
- provide missing code instructions (code instructions are provided with some instructions or code elements missing) that learners need to complete
- translate verbal/written instructions (algorithm) to code.
- add some functionality/instructions to an existing program.
- rewrite a set of coding instructions to be more efficient, e.g. using a loop construct for code that is repeated
- choose the correct solution from 2-3 options
- compare different solutions to evaluate efficiency
- debug an algorithm or program (find the bug, describe the bug and correct it)
- develop a solution/algorithm (code instructions) based on a given problem or for an open-ended problem through planning, implementing, testing and debugging.

depending on the competency/(ies) the learner needs to demonstrate.

Note:

Coding and Robotics, as a subject, in the Foundation Phase, is based on developing skills that underpin the processes of coding and robotics. To enable coding and robotics skills development it should be developed unplugged at first to reduce cognitive load and allow learners to focus on, and ground coding concepts. Schools that have educational programming or robotics tools and software could use these in combination with the unplugged approach. However, the curriculum is designed in such a way that it can be done unplugged throughout.

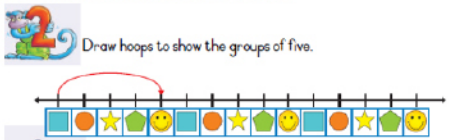
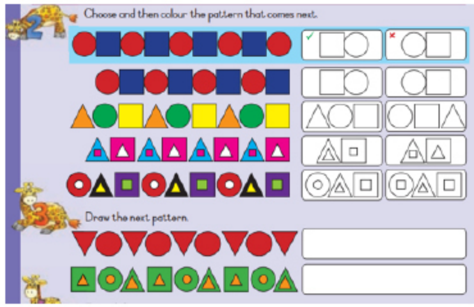
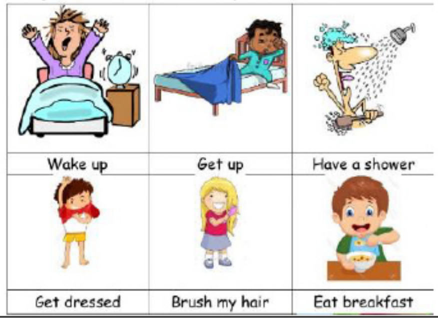
The table below describes the exit skills (shaded cells (per phase and per grade)).

2.15.1 Coding Competencies

Coding Content	Exit Skills per Grade			
Links to high level phase competencies (C.6, C.7)	**C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.** **C.7 Create or complete a pattern to represent a data set.**			
	GRADE R	**GRADE 1**	**GRADE 2**	**GRADE 3**
1. Pattern Recognition	**Exit Skills** to be mastered		**Prior knowledge must be covered in activities and progressed within the grade and across the phase.**	
1.1 Identify a pattern. **Identify a complete pattern present as a data set.**	1.1 Identify a simple pattern. (minimum 2 elements in core – repeated three times)	1.1 Identify a simple pattern. (minimum 3 elements in core – repeated three times)	1.1 Identify a more complex pattern. (minimum 3 elements in core – repeated four times)	1.1 Identify a more complex pattern. (minimum 4 elements in core – repeated four times)
1.2 Recognise pattern. **Recognise and interpret patterns in symbolic sets of data or visualisations.**	1.2 Recognise a simple type of pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.2 Recognise simple type of pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.2 Recognise more complex type of pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.2 Recognise more complex type of pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.3 Copy **Copy a pattern presented as a data set.**	1.3 Copy a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.3 Copy a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.3 Copy more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.3 Copy more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.4. Complete **Complete a pattern presented as a data set.**	1.4 Complete a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.4 Complete a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.4 Complete more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.4 Complete more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.5 Extend **Extend a pattern presented as a data set.**	1.5 Extend a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.5 Extend a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.5 Extend more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.5 Extend more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.6 Describe **Describe a pattern presented as a data set.**	1.6 Describe a simple pattern (depends on context) e.g. (colour, shapes texture, rhythm, sounds, movement)	1.6 Describe a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.6 Describe more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.6 Describe more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.7 Explain **Explain a pattern presented as a data set.**	1.7 Explain more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.7 Explain a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.7 Explain more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.7 Explain more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.8 Create **Create a pattern presented as a data set.**	N/A	1.8 Create a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.8 Create a more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.8 Create a more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)

1.9 Debugging **Debug a pattern presented as a data set.**	1.9 Debugging a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement) (Identify and correct one error <u>in the core - two elements</u> of a pattern that was repeated wrongly)	1.9 Debugging a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement) (Identify and correct one error <u>in the core -three elements</u> of a pattern that was repeated wrongly)	1.9 Debugging a more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement) (Identify and correct one error <u>in the core -three elements</u> of a pattern that was repeated wrongly)	1.9 Debugging a more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement) (Identify and correct one error <u>in the core -four elements</u> of a pattern that was repeated wrongly)
1.10 Compare **Compare patterns presented as a data set.**	1.10 Compare a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement) (Two patterns <u>in the core - two elements</u> differ with one element)	1.10 Compare a simple pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.10 Compare a more complex patterns (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)	1.10 Compare a more complex pattern (depends on context) e.g. (colour, shapes, texture, rhythm, sounds, movement)
1.11 Order / Sequence **Order/Sequence a pattern presented as a data set.**	1.11 Arrange a data set in a logical order to complete a task (set of two-three items/steps)	1.11 Arrange a data set in a logical order to complete a task (set of two-three items/steps)	1.11 Arrange a data set in a logical order to complete a task (set of three-five items/steps)	1.11 Arrange a data set in a logical order to complete a task (set of five-seven items/steps)

Examples of patterns:

Grade R	Grade 1	Grade 2	Grade 3
 https://www.fairypoppins.com/patterning-activities/	 DBE Grade 1 Workbook 2 (Maths) – p38	 DBE Grade 2 Workbook 1 (Maths) – p58	Daily behaviour routine pattern 

Note:
Pattern Recognition helps programmers to refine algorithms when developing coding solutions e.g., identifying repeating instruction patterns to be placed in loop constructs.
Also, pattern recognition leads to analysing patterns in data. By identifying patterns, we can predict what will come next and what will happen again and again in the same way and helps to make generalisations. A pattern may be numerical, visual or behavioural.

Coding Content	Exit Skills per Grade			
Links to high level phase competencies C.1, C.2, C.3, C.4, C.5 (Also links to C.6 and C.7)	**C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.** **C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs (Refer to next table – Implement the algorithm).** **C.3 Interpret and execute a given symbolic or written set of commands** **C.4 Debug a given symbolic or written set of instructions.** **C.5 Evaluate a given solution towards potential improvement.**			
	GRADE R	**GRADE 1**	**GRADE 2**	**GRADE 3**
2. Algorithm Design	**New skills to be mastered**		**Prior knowledge must be covered in activities and progressed within the grade and across the phase.**	
2.1 Listen to / Read problem statement	2.1 Listen to a simple problem statement or instruction.	2.1 Listen to more complex problem statement or instructions.	2.1 Listen to / Read more complex problem statement or instructions.	2.1 Listen to / Read more complex problem statement or instructions.
2.2 Understand the problem. (Visualise the problem)	2.2 Respond orally / kinaesthetically / pictorially to show understanding.	2.2 Respond orally / kinaesthetically / pictorially to show understanding.	2.2 Respond orally / kinaesthetically / pictorially / symbolically to show understanding.	2.2 Respond orally / kinaesthetically / pictorially / *symbolically/ written steps to show understanding.
2.3 Sequencing (Algorithm development) **Solve the problem**	2.3a Abstraction: Outline main ideas. 2.3b Decomposition: Illustrate / unpack steps to solve a problem. (e.g., objects, shapes, colours, pictures) (Minimum of 2-3 steps) depending on the problem. 2.3c Sequencing of the steps (i.e., Algorithm) The solution may be presented partially requiring the learners to complete it. (Maximum of 1 missing instruction) Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment/simulation scenarios.	2.3a Abstraction: Outline main ideas. 2.3b Decomposition: Illustrate / unpack steps to solve a problem. (e.g., objects, shapes, colours, pictures) (e.g., objects, shapes, colours, pictures, directional symbols) (minimum of 4-5 steps) depending on the complexity of problem. 2.3c Sequencing of the steps (i.e., Algorithm) The solution may be presented partially requiring the learners to complete it. (Maximum of 2 missing instructions) Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment/simulation scenarios.	2.3a Abstraction: Outline main ideas. 2.3b Decomposition: Illustrate / unpack steps to solve a problem. (e.g., objects, shapes, colours, pictures) (e.g., objects, shapes, colours, pictures, directional symbols, and alphabet letters) (minimum of 5-6 steps) depending on the complexity of problem. 2.3c Sequencing of the steps (i.e., Algorithm) The solution may be presented partially requiring the learners to complete it. (Maximum of 5 missing instructions) Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment/simulation scenarios.	2.3a Abstraction: Outline main ideas. 2.3b Decomposition: Illustrate / unpack steps to solve a problem. (e.g., objects, shapes, colours, pictures) (e.g., of objects, shapes, colours, pictures, directional symbols, alphabet letters and short sentences) (minimum of 6-7 steps) depending on the complexity of problem. 2.3c Sequencing of the steps (i.e., Algorithm) The solution may be presented partially requiring the learners to complete it. Problems could include: • Grid-based scenarios • Story-based scenarios • Movement-based scenarios (e.g., dance moves) • Robot enactment/simulation scenarios.
2.4 Visual / written representation (Create the code)	2.4 Presenting a visual solution/algorithm with objects, arrows, or pictures.	2.4 Presenting a visual solution/algorithm with objects, arrows, or pictures.	2.4 Presenting a visual solution/algorithm with objects, arrows, pictures, symbols, or words.	2.4 Presenting a visual solution/algorithm with objects, arrows, pictures, symbols, or short written commands.
2.5 Implementation: Execute-Test-Debug (Adjustments applied until outcomes reached) In person (acting out) or on paper (tracing)	2.5a Implement the solution/algorithm. (A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool)	2.5a Implement the solution/algorithm. (A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool)	2.5a Implement the solution/algorithm. (A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool)	2.5a Implement the solution/algorithm. (A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool)

Find errors and correct	2.5b Test: Did the implementation reach the criteria? 2.5c Trace the error (determine the cause) 2.5d Debug to find problem in solution/algorithm and do correction.	2.5b Test: Did the implementation reach the criteria? 2.5c Trace the error (determine the cause) 2.5d Debug to find problem in solution/algorithm and do correction.	2.5b Test: Did the implementation reach the criteria? 2.5c Trace the error (determine the cause) 2.5d Debug to find problem in solution/algorithm and do correction.	2.5b Test: Did the implementation reach the criteria? 2.5c Trace the error (determine the cause) 2.5d Debug to find problem in solution/algorithm and do correction.
2.6 Evaluate (Determine whether the solution solved the problem or which solution is better/best)	2.6 Evaluate: Did the solution meet the criteria to solve the problem?	A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool.	A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool.	A simple set of commands in relation to the designed algorithm are correctly executed physically, on paper or with an educational tool.
2.7 Compare and reflect (Learn from all solutions)	2.7a Compare different solutions to identify the different approaches. 2.7b Reflect and find the most optimal solution. (The evaluation and comparison relate to the solution/s made by all the learners)	2.7a Compare different solutions to identify the different approaches. 2.7b Reflect and find the most optimal solution. (The evaluation and comparison relate to the solution/s made by all the learners)	2.7a Compare different solutions to identify the different approaches. 2.7b Reflect and find the most optimal solution. (The evaluation and comparison relate to the solution/s made by all the learners)	2.7a Compare different solutions to identify the different approaches. 2.7b Reflect and find the most optimal solution. (The evaluation and comparison relate to the solution/s made by all the learners)

Examples

Example 1
See problem-solving example on page 10/11.

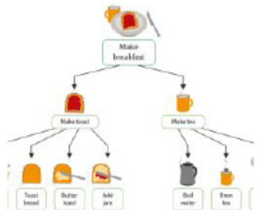
Example 2
See example Section 3 Grade 3 Term 4 on page 117/118

Computational Thinking Examples

Abstraction

A world map is an abstraction of the earth in terms of longitude and latitude, helping us describe the location and geography of a place

Decomposition



Break up the task of making breakfast in several smaller tasks:
Make toast
Bake egg
Make coffee

Pattern Recognition

Irregular heartbeat can be identified looking at deviations from the normal pattern.

This can help to diagnose medical conditions

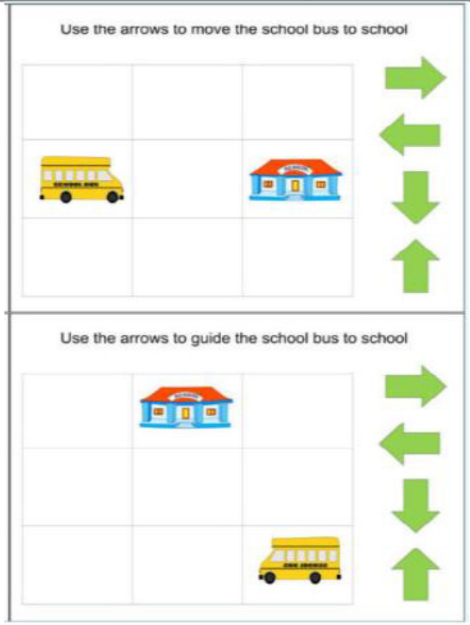
Algorithm

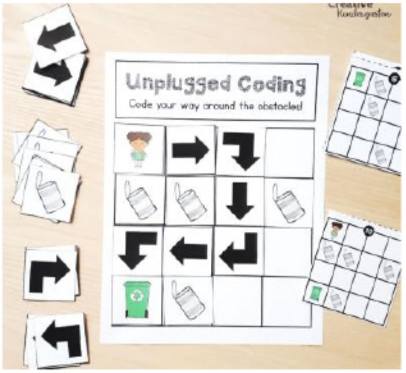
Steps/Instructions for making toast (1 subtask in making breakfast)

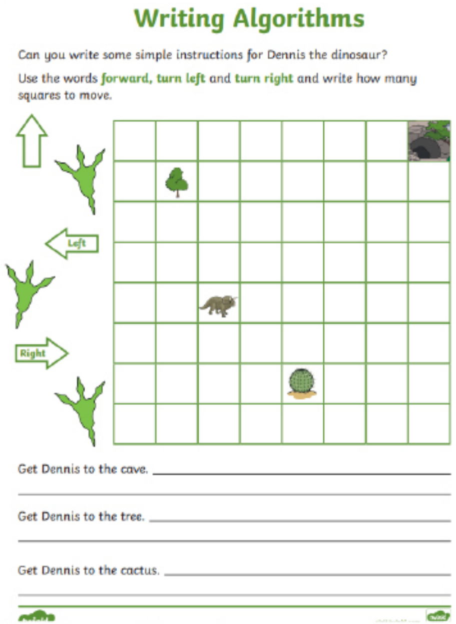
Coding Content	Exit Skills per Grade			
Links to high level phase competencies C.1, C.2	**C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.** **C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands single repetition and conditional constructs.**			
	GRADE R	**GRADE 1**	**GRADE 2**	**GRADE 3**
3. Coding:	**New skills to be mastered**		**Incidental learning**	
3.1 Event trigger (Start)	3.1 Specific instruction on to trigger event (start an action) and an end set of commands. (Place- e.g., beacon/flag, beginning of an activity- count/whistle).	3.1 Specific instruction on to trigger event (start an action) and an end set of commands. (Place- e.g., beacon/flag, beginning of an activity- count/whistle).	3.1 Specific instruction on to trigger event (start an action) and an end set of commands. (Place- e.g., beacon/flag, beginning of an activity- count/whistle).	3.1 Specific instruction on to trigger event (start an action) and an end set of commands. (Place- e.g., beacon/flag, beginning of an activity- count/whistle).
3.2 Instruction: **Apply computational thinking skills to develop a set of logical instructions (algorithm) to solve a problem.** **(See table 2) – Algorithm design.**	**3.2 Sequence (Instruction set):** Simple operations are presented using sequences of pictures (Maximum**1 up to 4) Logically order a set of pictures to accomplish a set task. (Maximum up to 4) Order, arrange or search a set of pictures and symbols according to given criteria. Follow basic instructions (until barrier is hit – not block-by-block) using objects, arrows, and pictures: - Forward / Backward - Up / Down - Any other single action (Incidental) (Turn left /Turn right is implied – by implication, learners turn to face the direction they will move in next)	**3.2 Sequence (Instruction set):** Simple operations are presented using sequences of pictures and or simple three-word sentences. (Maximum up to 5) Logically order a set of pictures or three-word sentences to accomplish a set task. (Maximum up to 5) Order, arrange or search a set of pictures and symbols, characters, and numbers according to given criteria. Follow simple instructions using a grid, objects, arrows, or pictures: - Forward / Backward - Up / Down - Over/Under - Right / Left (Implied/Incidental) - (Any single command, e.g., Jump, Turn around, etc.)	**3.2 Sequence (Instruction set):** Simple tasks as logical instructions are identified to solve a problem. (Maximum up to 7) Simple operations are presented using sequences of pictures and or simple sentences. (Maximum up to 7) Logically order a set of pictures simple sentences to accomplish a set task. (Maximum of 7) Order, arrange or search a set of pictures, symbols, characters, numbers, and words according to given criteria. Decompose more complex instructions using a grid, objects, arrows, or pictures: - Forward / Backward - Right / Left / Turn around - Up / Down - Jump over / Shoot - Grab / Release - Pick up / Put down	**3.2 Sequence (Instruction set):** Simple tasks as logical instructions are identified to solve a problem from which unnecessary or irrelevant details are ignored. (Maximum up to 9) Simple operations are presented using sequences of pictures and or simple sentences. (Maximum up to 9) Logically order a set of pictures simple sentences to accomplish a set task. (Maximum up to 9) Order, arrange or search a set of pictures, symbols, characters, numbers, and words or sentences according to given criteria. Decompose more complex instructions using a grid, objects, arrows, or pictures: - Forward / Backward - Right / Left / Turn around - Up / Down - Jump over / Shoot - Grab / Release - Pick up / Put down
3.3 Decision (Condition): Incorporate decisions (conditional constructs) as part of the solution code.	**N/A**	**N/A**	3.3 Decision (Condition): Decide on an action based on a condition. E.g., Purple /orange carrot	3.3 Decision (Condition): Decide on an action based on a condition. E.g. Purple /orange carrot
3.4 Repetition: **Incorporate repetitions (loop concepts) as part of the solution code.**	**3.4 Repetition (Loop concept):** Repeating an action, instruction. (Incidental learning)	**3.4 Repetition (Loop concept):** Repeating an action, instruction.	**3.4 Repetition (Loop concept):** Repeating an action, instruction.	**3.4 Repetition (Loop concept):** Repeating an action, instruction.
3.5 Input-Processing-output: **Demonstrate how input and processing result in output.** **Debug (reflecting on) a given symbolic or written set of instructions.**	3.5 **Implement: Problem-solving**: Algorithm design process here. (Incidental learning)	3.5 **Implement: Problem-solving**: Algorithm design process here. (Incidental learning)	3.5 **Implement: Problem-solving:** Algorithm design process here.	3.5 Implement: Problem-solving: Algorithm design process here.

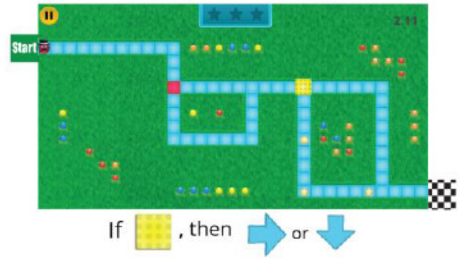
Examples:

Example 1

<table>
<tr><th>Grade R</th><th>Grade 1</th><th>Grade 2</th><th>Grade 3</th></tr>
<tr><td colspan="4">C.2 – Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs</td></tr>
<tr>
<td>

https://funstemlab.com/unplugged-coding-activities-7-free-printable-coding-worksheets/</td>
<td>

https://creativekindergartenblog.com/unplugged-coding-for-kindergarten/</td>
<td>

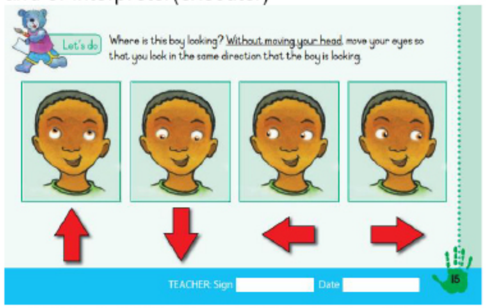
https://www.twinkl.co.za/</td>
<td>

If , then or
https://www.kodable.com/hour-of-code/unplugged-coding</td>
</tr>
</table>

Note

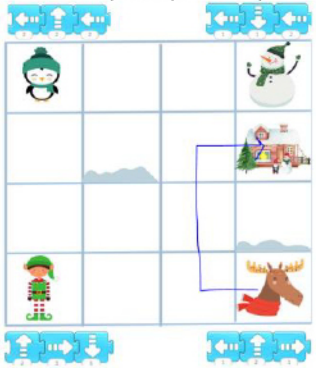
A **solution (set of instructions/algorithm)** can include coding constructs such as repetition (loops) and decisions (IF…THEN)

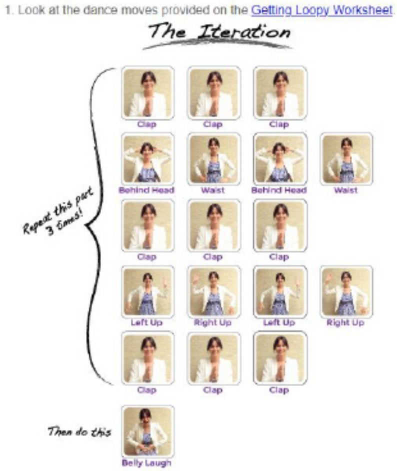
Example 2:

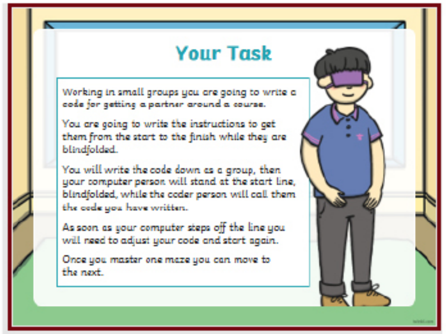
<table>
<tr><th>Grade R</th><th>Grade 1</th><th>Grade 2</th><th>Grade 3</th></tr>
<tr><td colspan="4">C.3 – Interpret and execute a given symbolic or written set of commands (algorithm)</td></tr>
<tr>
<td>One learner could take on the role of instructor and or interpreter(executer)


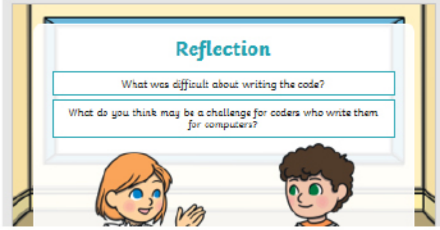
DBE Grade R Workbook 1 (English) – p15

Game based rules and commands
A set of game rules are nothing more than an algorithm (instructions to execute). The game of match two cards or snap can easily be presented and conceptualised. This in terms of the teacher and not so much the learners.


DBE Grade R Workbook 2 (English) – p54
The rules of snap are simple, the same for the memory game.</td>
<td>One learner could take on the role of instructor and or interpreter(executer)

Code a classmate

https://www.twinkl.co.za/</td>
<td>One learner could take on the role of instructor and or interpreter(executer)


https://code.org/curriculum/course1/12/Teacher#Activity1</td>
<td>One learner could take on the role of instructor and or interpreter(executer)




https://www.twinkl.co.za/</td>
</tr>
</table>

Note:
It is important to note that there will always be a degree of overlap between Coding content and Robotics (R.7) content, e.g. developing algorithms and coding instructions for the Coding strand and developing algorithms and code instructions in the Robotics strand (R.7) as programming (coding) concepts are applied to control robots to perform specific tasks.

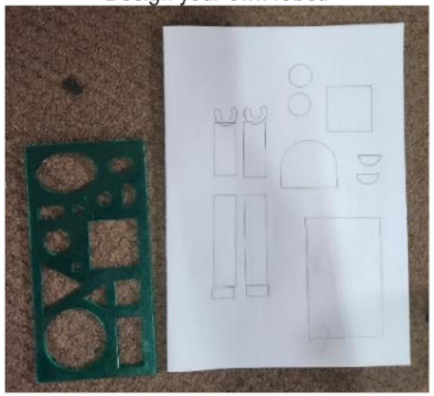
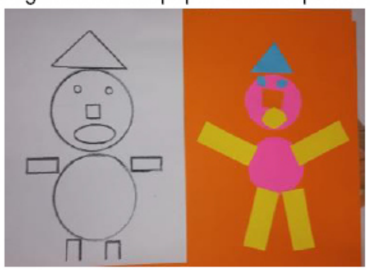
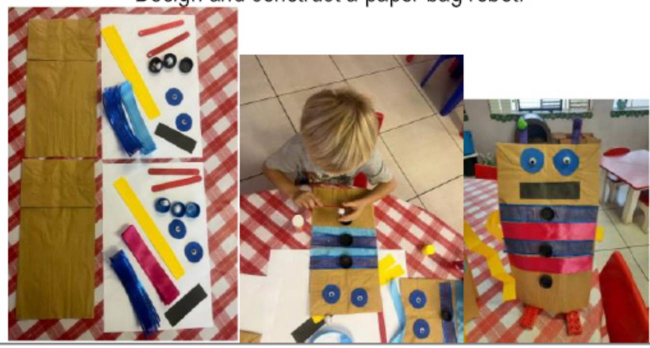
2.15.2 Robotics Competencies

Robotics Content	Exit Skills per Grade			
Links to high level phase competencies **R.1, R.2, R.3, R.4, R.5, R.6, R.7**	**R.1 Explain what a robot is in simple terms.** **R.2 Identify different types of robots.** **R.3 Outline the different components of a robot** **R.4 Present an understanding of how robots affect the world.** **R.5 Design a simple product (artefact) based on a set of design specifications.** **R.6 Mimic the operations of a robot.** **R.7 Create test and execute a set of robotic instructions.**			
	GRADE R	**GRADE 1**	**GRADE 2**	**GRADE 3**
4. Robotic Skills	**New skills to be mastered**		**Incidental learning**	
4.1 Identify types of robots **Identify different types of robots.**	4.1 Identify different types of robots	4.1 Different examples of robots and what they do are listed.	4.1 The identification classification relates to robots that are used in factories and robots that are not used in factories (Service robots)	4.1 The identification classification includes domestic and professional use robots.
4.2 Explain - What robots are? **Provide an elementary explanation of what a robot is.**	4.2 A rudimentary explanation of what a robot is, is presented.	4.2 A basic explanation of what a robot is, is presented including reference to their purpose.	4.2 A basic explanation of what a robot is, is presented including reference to their purpose and mode of operation. Reference to moving and sensory parts are made.	4.2 A basic explanation of what a robot is, is presented including reference to their purpose and mode of operation. Reference to moving and sensory and processing parts are made.
4.3 Explain - How they work? **Outline the different components of a robot.**	4.3 The learners present the concept that a robot comprises of different components each with a purpose (incidental learning).	4.3 The learners present the concept that a robot comprises of different components each with a purpose.	4.3 The learners present the concept that a robot comprises of different components each with a purpose. Reference is made to a power source and motors.	4.3 The learners present the concept that a robot comprises of different components each with a purpose. Reference is made to the following concepts as part of the outline: • Robots comprise of mechanical parts. • Requires power. • Require some form of programming.
4.4 What can robots do? **Explain what a robot is in simple terms and what are they used for**	4.4 A rudimentary explanation of what robots are used for is given.	4.4 A basic explanation of what robots are used for is given that references specific tasks.	4.4 A basic explanation of what robots are used for is given that references specific tasks including dangerous and repetitive ones.	4.4 A basic explanation of what robots are used for is given that references specific concepts that robots can be programmed to react to their environment.
4.5 Act as robots **Mimic the operations of a robot from a set of instructions** **Relates and links to** **2. Algorithm design and** **3. Coding for the grade.** **(The instructions are given, and the learners need to execute the instructions physically or on paper, or using body coding (or optional, an educational tool))**	4.5 The learners mimic the operations of a robot based on given instruction or for a purpose. Simple instructions are performed, in person or using a tool, on paper.	4.5 The learners mimic the operations of a robot based on given instruction or for a purpose. Simple instructions are performed, in person or using a tool.	4.5 The learners mimic the operations of a robot based on given instruction or for a purpose. Instructions are performed, in person or using a tool.	4.5 The learners mimic the operations of a robot based on given instruction or for a purpose. Instructions are performed, in person or using a tool.

4.6 Learners develop code instructions based on an algorithm. **Create, test, and execute a set of robotic instructions.** **Relates and links to:** **2. Algorithm design and** **3. Coding for the grade.**	4.6 A rudimentary set of instructions are compiled and executed to perform a task.	4.6 A rudimentary set of instructions are compiled and executed to perform a task.	4.6 A rudimentary set of instructions are compiled and executed to perform a task.	4.6 A rudimentary set of instructions are compiled and executed to perform a pr task.
4.7 Learners develop more complex code instructions based on an algorithm. **Present a coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.**	4.7 Symbols are used to represent actions and operations to accomplish a task. Each symbol represents a single task. (**Minimum of 3 different actions/symbols) (Maximum of 6 steps) (Incidental)	4.7 Symbols are used to represent actions and operations to accomplish a task. Symbols may be grouped to represent repetition. (**Maximum of 5 different actions/symbols) (Maximum of 8 steps) (Incidental)	4.7 Symbols (normal or block (puzzle) type) or written statements are used to represent actions and operations to accomplish a task. Symbols / blocks may be grouped to represent repetition. Symbols / blocks may be used to represent a condition. (**Maximum of 5 different actions/symbols) (Maximum of 10 steps) Symbols may include block-code type images with linkages.	4.7 Symbols (normal or block (puzzle) type or written statements are used to represent actions and operations to accomplish a task. Symbols / blocks may be grouped to represent repetition (or a statement indicating repetition) Symbols / blocks may be used to represent a condition (or a statement indicating condition) (**Maximum of 6 different actions/symbols) (Maximum of 14 steps) Symbols may include block-code type images with linkages.
4.8 Design and make: **Design a simple artefact based on a set of design specifications.**	4.8 A simple artefact is created to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk about what their robots can do.	4.8 A simple artefact is created to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk about what their robots can do. Strings / sticks and or pins may be added to mimic movement. Different materials can be used, e.g., Pipe cleaners, ice cream sticks, straws, recycled materials (toilet rolls, lids, pill cases, egg containers) etc. (Life Skills-Art) The creation of the artefact could also take on the form of a game e.g. (Assemble by numbers) (Life Skills- Physical Ed)	4.8 A simple artefact is created to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk about what their robots can do. Strings and or pins may be added to mimic movement. Different materials can be used, e.g., Pipe cleaners, ice cream sticks, straws, recycled materials (toilet rolls, lids, pill cases, egg containers) etc. The creation of the artefact could also take on the form of a game e.g. (Assemble by numbers) Throw some dice. (Mathematics) Assembly using prefabricated parts if (available) e.g., Building blocks.	4.8 A simple artefact is created to represent a robot or equivalent. Step by step instructions can be applied or given. The activity may be open where various materials are supplied to the learners to have them create their own robot and/or related artefact. The learners reflect and talk about what their robots can do including the composition of the various parts and the purpose of each. Strings and or pins or lever mechanisms and or pullies may be added to mimic movement. Different materials can be used, e.g., Pipe cleaners, ice cream sticks, straws, recycled materials (toilet rolls, lids, pill cases, egg containers) etc.

			The instructions contain various steps that should be read and or interpreted as part of the assembly. (Lang) The assembly should require a set order (one step should follow the other)	The creation of the artefact could also take on the form of a game e.g. (Assemble by numbers) Throw some dice. Assembly using prefabricated parts if (available) e.g., Building blocks. The instructions contain various steps that should be read and or interpreted as part of the assembly. The assembly should require a set order (one step should follow the other)

Examples:

Grade R	Grade 1	Grade 2	Grade 3
Patterns and a corresponding artefact. 	Design your own robot. 	Robot with paper folding and other items. 	Robot hand with moving parts. 
Design a robot on paper with shapes and  constructing the robot afterwards.	Design and construct a paper bag robot. 		

2.15.3 Digital Concepts Competencies

Digital Concepts	Exit Skills per Grade			
Links to high level phase competencies **D.1, D.2, D.3, D.4, D.5, D.6, D.7, D.8, D.9**	**D.1 Outline the concept of technology and purpose of information technology (IT).** **D.2 Recognise that he or she is living as citizens in a digital world.** **D.3 Demonstrate an understanding of the concept of a computing device.** **D.4 Identify the common uses of ICT in the real world.** **D.5 Differentiate between the components of an ICT system.** **D.6 Explain how the adaption of technology impacted the world we work and live in.** **D.7 Present a basic understanding of the concept of input processing and output.** **D.8 Interpret a pattern to represent or communicate a message or image.** **D.9 Create a pattern to represent or communicate a message or image.**			
5. Digital Concepts	**Exit Skills** to be mastered		**Prior knowledge must be covered in activities and progressed within the grade and across the phase.**	
5.1 Recognise **Outline the concept of technology and purpose and identify the common uses of Computing Device in the real world.**	**N/A**	5.1 A rudimentary list of the use of IT related technologies and devices are named in terms of their use. (Cell phone, laptop, smart TV) (Incidental learning) Links with D.1 and D.2	5.1 An elementary list of the use of IT related technologies and devices are named in terms of their use. Links with D.2	5.1 An elementary list of the use of IT related technologies and devices are named in terms of their use. Links with D.1.3 and D.2.3
5.2 Identify **Differentiate between the components of a Computing System by interpreting patterns to represent or communicate a message or image.**	5.2 A rudimentary explanation of what technology is, is presented. (Incidental learning) Learners can point out examples of technology. (Incidental learning)	5.2 A rudimentary explanation of what technology is, is presented. Learners can point out examples of technology and relate its use to everyday life. Learners relate the concept of technology to that of an electronic device.	5.2 A rudimentary explanation of what technology is, is presented. Learners can point out examples of technology and relate its use and purpose to everyday life. Learners relate the concept of technology to that of an electronic device.	5.2 A rudimentary explanation of what technology is, is presented. Learners can point out examples of technology and relate its use and purpose to everyday life. Learners relate the concept of technology to that of an electronic device. The students answer includes that the technological artefact has a common or specific goal. The answer also includes the concept that technologies often comprise of different components.
5.3 Operating **Demonstrate an understanding of the concept of a computing device.**	5.3 A rudimentary explanation of what an electronic device is, is presented. (Incidental learning) Learners can point out examples of electronic devices.	5.3 An elementary explanation of what an electronic device is, is presented. Learners can point out examples of electronic devices. The learners answer should incorporate the concept that an electronic device can follow and interpret instructions. Links with D.1	5.3 An elementary explanation of what an electronic device is, is presented. Learners can point out examples of electronic devices. The learners answer should incorporate the concept that an electronic device can follow and interpret instructions. Links with D.1	5.3 An elementary explanation of what an electronic device is, is presented. Learners can point out examples of electronic devices. The learners answer should incorporate the concept that an electronic device can follow and interpret instructions and produce output/result or render an outcome. Links with D.1

5.4 Create **Create a pattern to represent or communicate a message or image.**	N/A	5.4 A basic pattern is created to represent an image or communicate a message or an image. A basic pattern is encoded to a simple word, image -to symbols, or 3-word maximum phrase. Done in relation to C.6 and D.8	5.4 A basic pattern is created to represent an image or communicate a message or an image. A basic pattern is encoded to a simple word, image -to symbols, or simple sentence. Done in relation to C.6.and D.8.	5.4 A basic pattern is created to represent an image or communicate a message or an image. A basic pattern is encoded to a simple word, image -to symbols, or simple sentence. Done in relation to C.6 and D.8
5.5 Apply **Present a basic understanding of the concept of input, processing and output by demonstrating a basic proficiency in the application of digital skills.**	5.5 The learners present an understanding that input results in some form of output.	5.5 The learners present an understanding that input results in some form of output. Input Instructions are executed those results in an action. The concept that different forms of input results in different actions are emphasised.	5.5 The learners present an understanding that input results in some form of output. Input Instructions are executed those results in an action. Output as a form of communication from the device The concept that different forms of input results in different actions are emphasised.	5.5 The learners present an understanding that input results in some form of output. Input Instructions are executed those results in an action. Output as a form of communication from the device The concept that different forms of input results in different actions are emphasised. The concept that processing takes place between input and output forms part of the learners understanding.
5.6 Digital Citizenship **Recognise that he or she is living as citizens in a digital world by explaining how the adaption of technology impacted the world, we work and live in.**	5.6 The learners present an understanding that the digital world is all around us. The learners understand that electrical devices (Dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and use in moderation (screen time)	5.6 The learners present an understanding that the digital world is all around us. The learners understand that electrical devices (Dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and use in moderation (screen time). The conceptualisation is presented in terms of D.1	5.6 The learners present an understanding that the digital world is all around us. The learners understand that electrical devices (Dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and use in moderation (screen time). The learners understand that protecting information with a password helps keep it private. The concept of a digital footprint is also introduced at an elementary level. The conceptualisation is presented in terms of D.1	5.6 The learners present an understanding that the digital world is all around us. The learners understand that electrical devices (Dangers of electricity) should be used safely (e.g., don't use electronic devices whilst crossing the street) and use in moderation (screen time). Present an understanding of the dangers of going online. Present a basic understanding of the concept of cyberbullying and how to deal with it. The learners understand that protecting information with a password helps keep it private. The concept and dangers of sharing information like personal information usernames and or passwords are recognised. The responsible use of technology is referenced as part of the concept. The concept of a digital footprint is also introduced at an elementary level. The learners present an understanding of the necessary to report unsuitable use of

				electronic communication, the access of content and or contact. The conceptualisation is presented in terms of D.1
5.7 Patterns as data (Data representation)	N/A	5.7a Interpret a pattern to represent or communicate a message or image. A basic pattern is interpreted and a corresponding message in symbolic form is presented. (Incidental) 5.7b Create a pattern to represent or communicate a message or image. A basic pattern is created to represent an image or communicate a message or an image. (Incidental)	5.7a Interpret a pattern to represent or communicate a message or image. A basic pattern is interpreted and a corresponding message in symbolic form is presented. 5.7b Create a pattern to represent or communicate a message or image. A basic pattern is created to represent an image or communicate a message or an image.	5.7a Interpret a pattern to represent or communicate a message or image. A basic pattern is interpreted and a corresponding message in symbolic form is presented. 5.7b Create a pattern to represent or communicate a message or image. A basic pattern is created to represent an image or communicate a message or an image.

Note:

Most of the Digital Concepts content could be integrated with aspects of Coding or Robotics content, e.g. if learners are working on a device, learners could be asked to demonstrate an understanding of the concept of a computing device (5.3), referring to their uses, components and the concept of input-processing-output.

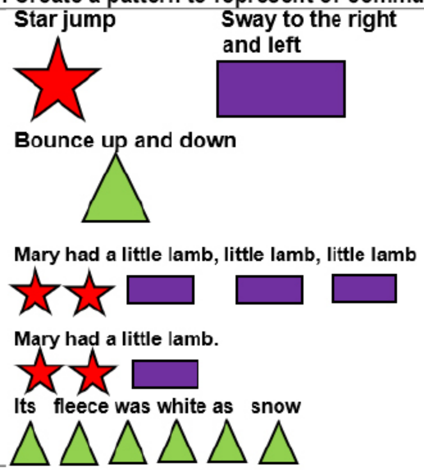
Examples:

Example 1 (Patterns as data (5.7))

Grade R	Grade 1	Grade 2	Grade 3

D.9: Create a pattern to represent or communicate a message or image.

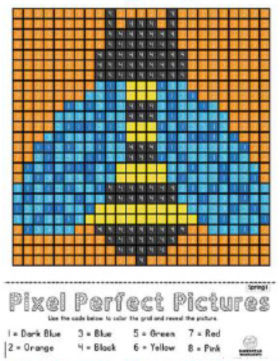
Grade R



Grade 1

A basic pattern is encoded to a simple word, image, or 3-word maximum phrase.

Done in relation to C.6 and D.8

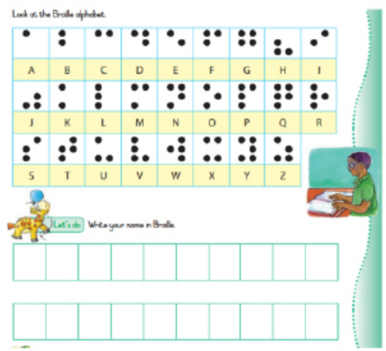


https://superstarworksheets.com/logic-worksheets/pixel-coloring-pages/

Grade 2

A basic pattern is encoded to a simple word, image, or simple sentence.

Done in relation to C.6.2 and D.8.2.

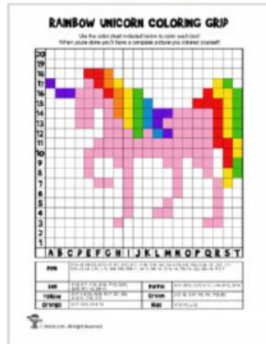


DBE Grade 2 Workbook 2 (Life skills) – p52

Grade 3

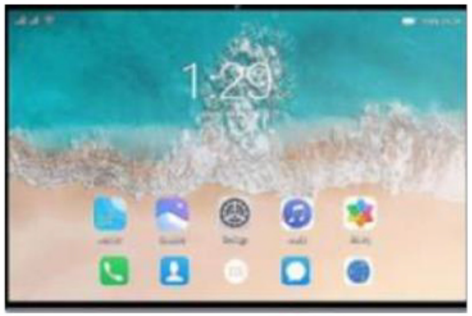
A basic pattern is encoded to a simple word, image, or simple sentence.

Done in relation to C.6 and D.8D



https://www.woojr.com/mystery-picture-grid-coloring-pages-fantasy-fairy-tales/

Example 2 (Digital Technologies with input that results in output and various components such as touch screen (input/output), remote (TV)))

Grade R	Grade 1	Grade 2	Grade 3
D.4 Identify the common uses of ICT in the real world. **D.5 Differentiate between the components of an ICT system.** **D.7 Present a basic understanding of the concept of input processing and output.**			
 Mobile Phone Technology with concept of input that results in output	Tablet – concept of input that results in output 	Laptop with keyboard (input) or mouse (input) and screen (output) 	Smart TV with remote control device (input) 

3 SECTION 3 CONTENT SPECIFIC CLARIFICATION PER GRADE PER TERM

The following tables provide the content clarification per term and per grade.

This section should be read in conjunction with Figure 2.6, Table 2.2, Table 2.3 and Table 2.4

In Foundation Phase, the curriculum is designed to integrate with other Foundation Phase subjects as indicated in the term plans. This integration could also strengthen the specific concepts and content in the subject it is integrated with.

Content clarification is done with examples as Coding and Robotics is a new subject.

Note:

This section contains examples that clarify the content and competencies. These examples serve as illustrations to better understand the topics and the abilities students are expected to develop.

However, teachers should see these examples as a starting point for teaching the content and competencies. While the examples are beneficial, teachers should not limit themselves to just those activities. They are encouraged to include other exercises and tasks to ensure deliberate practise and a deeper understanding of the concepts and skills being taught.

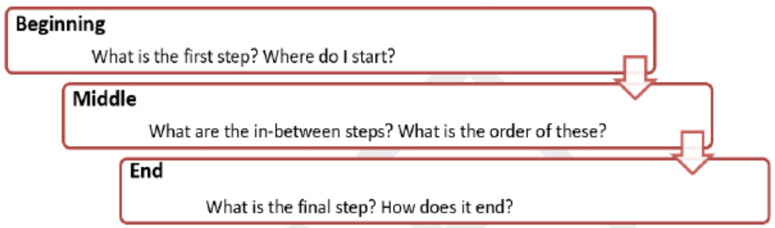
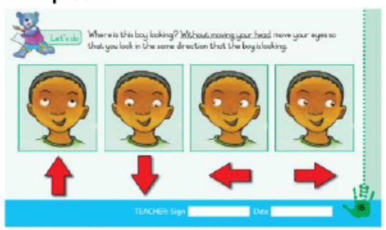
The content and competencies are also grouped based on the main topic areas. This organisation helps teachers understand which skills and knowledge are related and how they are connected. The content and competencies are therefore not necessarily listed in the order they must be taught. Teachers have flexibility in how they sequence the topics based on the context of their teaching environment and the needs of their leaners. However, there is an indication of how different competencies relate to each other. This linkage could help teachers understand the progression of skills and how they support or build upon one another.

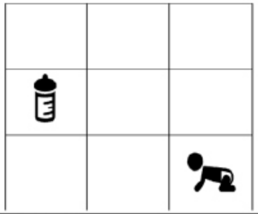
Teachers should therefore develop their Annual Teaching Plans (ATPs) sequencing content and competencies in a manner that will make sense for their learners and their teaching and learning environment to foster a positive learning experience. The goal of developing the ATPs is to maximize the learners' learning outcomes and achievement.

3.1 GRADE R

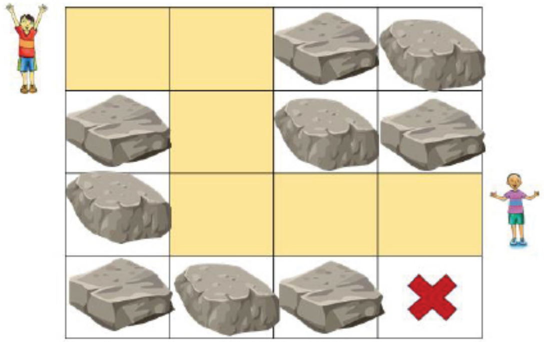
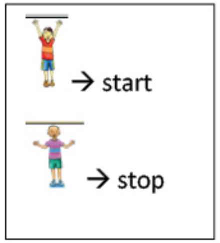
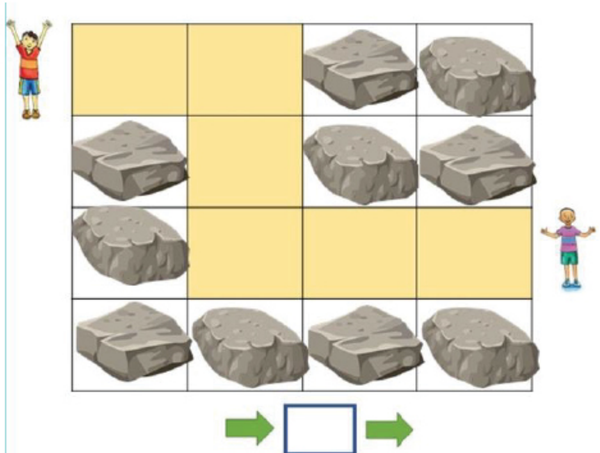
Note:
Teachers must include the following competencies and content in their Annual Teaching Plan (ATP), distributed across the terms and sequenced, organised and grouped (and integrated with other subjects in Grade R where applicable) in a manner that will facilitate learning ensuring ample retrieval and deliberate practise with feedback to ground principles and concepts, maximize the learners' learning outcomes and achievement but also ensure a gradual learning curve, and in a way that will make optimal use of time and resources.

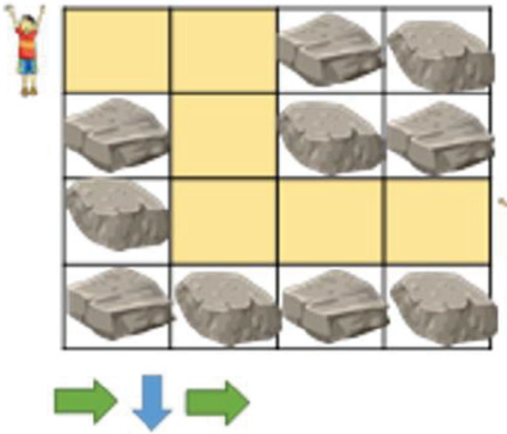
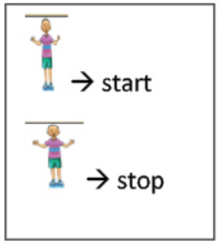
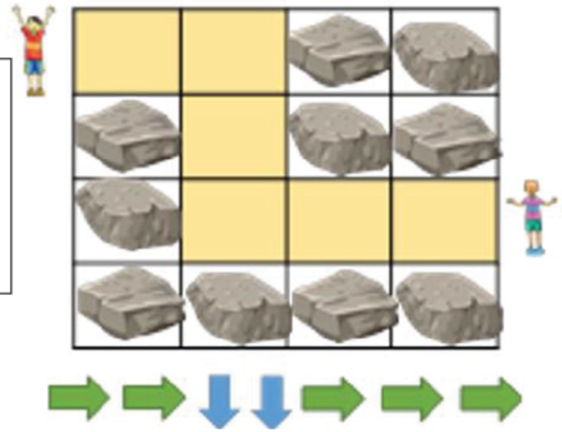
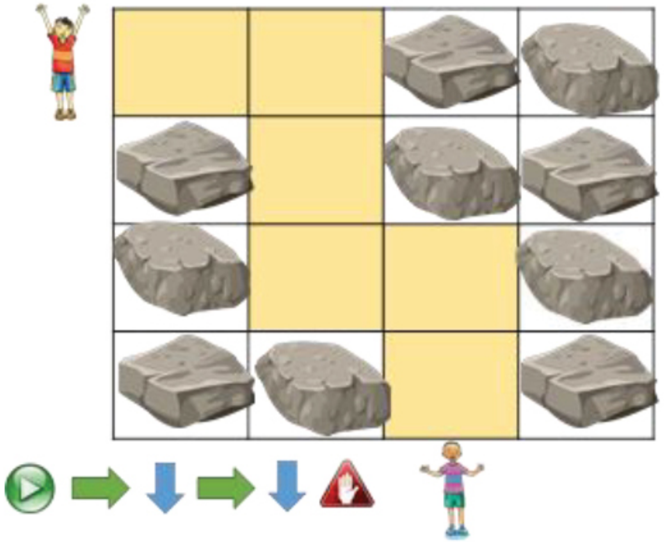
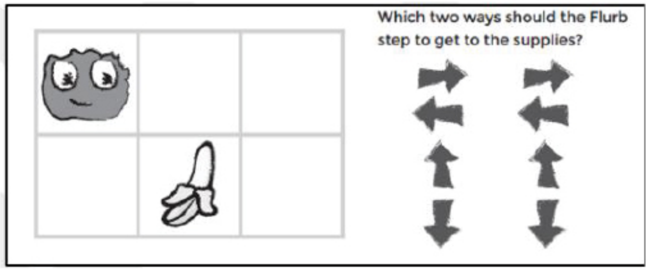
3.1.1 Term 1

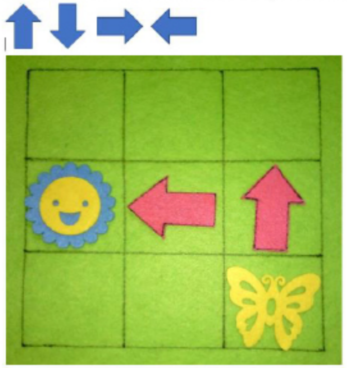
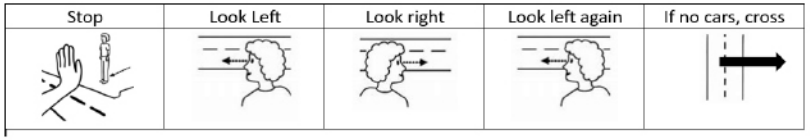
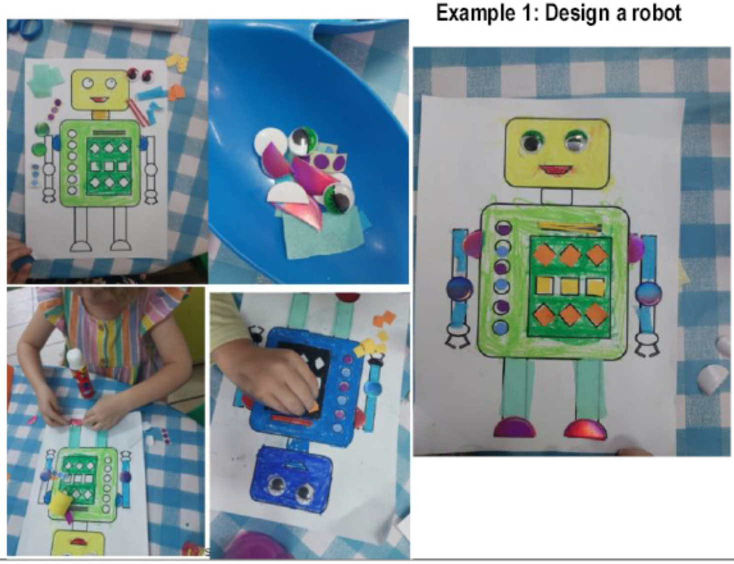
Content (Grade R / Term 1)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics**
C.6: Recognise and interpret patterns in symbolic sets of data or visualisations:	C.6 develops **pattern recognition** (part of computational thinking) that is eventually used to develop coding solutions as part o computational thinking to identify patterns in the coding problem and/or data by identifying similarities or differences that can help to solve the problem or refine the algorithm.
Identify a complete pattern presented as a data set **Examples:**  Learners need to understand that a pattern is a design that repeats.	
Algorithm Design and Coding:	**Algorithm Design and Coding mostly go together.**
C.1: Apply computational thinking skills to develop a set of logical instructions to solve a problem	**Could be integrated with Language**
Examples: **Picture Stories:**  Every sequence has a beginning, a middle and an end (just like a story).  	Introduce the concept of an algorithm (set of logical instructions or commands, carried out in a specific sequence) as well as the concept of a basic structure of a program (begin, middle, end). Picture stories are presented in a logical order (sequenced correctly with a beginning and an end). When ***sequencing***, we learn about patterns and relationships, and to understand the order of things. By learning to sequence, we develop the ability to understand and arrange purposeful patterns of actions, behaviours, ideas, or thoughts that supports the logical sequencing of coding instructions.
C.3: Interpret and execute a given symbolic or written set of commands	**Could be integrated with Language**
Examples:  DBE Grade R Workbook 1 (English) – p.15 **Note:** Learners do not need to understand the word algorithm at this stage. The concept of 'algorithm' is introduced as a set of logical instructions or commands that are carried out in a specific sequence and that a human or a computer can understand and follow/execute.	Provide learners with a set of symbolic instructions which they need to interpret and carry out. In coding, one develops sets of logically ordered instructions which the computer can understand and carry out. The computer can only understand instructions and follow them ***exactly*** the way they are presented and interpreted.

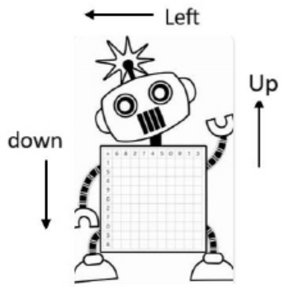
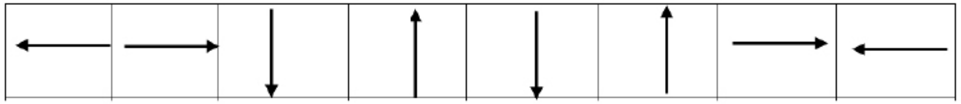
Content (Grade R / Term 1)	Notes/Examples
Robotics	**Could be integrated with Creative Arts**
R.1: Explain what a robot is in simple terms	**Could be integrated with Life Skills**
Activity Learners give their ideas of what a robot is. Techer then discusses these ideas and leads learners towards robot concepts. Teacher can show pictures of robots. robot - Kids | Britannica Kids | Homework Help	Eventually learners need to understand that a robot is a machine built by humans and programmed to perform tasks, which a human can also do, e.g. a robot vacuum cleaner. A robot can thus substitute a person, performing a task that the person could do. Robots can only do what they are programmed to do (follow a set of instructions).
R.2: Design a simple product (artefact) based on a set of design specifications	**Could be integrated with Life Skills /Creative Arts**
Examples: Identify the patterns used, then create an artefact with the same pattern: 	Link with pattern recognition (C.6) – learners first need to identify the pattern. Learners can use different patterns. Learners can afterwards identify each other's patterns. Eventually links to R.5 – learners design a 'robot' that can 'do' something.
R.6: Mimic the operations of a robot	**Links to C.3**
Examples: Help the baby find its bottle. Learners act out the three sets of instructions on a grid to see which set would get the baby to the bottle.  ❶ ↑← ❷ ←↑ One can extend the problem to move one block at a time, then more options can be provided, e.g.: ❶ ↑↑← ❷ ↑←← ❸ ←←←	Learners need to interpret and mimic (execute) the given symbolic instructions (set of arrow commands/instructions (algorithm)) One arrow implies movement until end of grid or object is hit (not one block at a time)
Digital Concepts	**Could be integrated with Life Skills**
D.2: Recognise that he or she is living as citizens in a digital world	**D.2 and D.3 can be done together.** Could be linked to when learners mimic the operations of a robot, explaining that a mobile phone needs instructions (input) from the user (interprets instruction – processing) to work (provide output). Aspects that could be discussed: • Safe use of electronic devices • Limited screentime
Limiting screentime: Explain to learners that they need to balance time spent on screens with other activities such as playing outside, spending time with family and friends, listening to stories (e.g. someone reading to them), etc.	
D.3: Demonstrate an understanding of the concept of a computing device	
Examples: As most people, today, use digital technologies to communicate, learn and work, emphasise the responsible and ethical use of digital technologies and online platforms as digital citizen. Provide examples of responsible use and behaviour. Ask questions about the digital technologies that learners are familiar with. Use, e.g. old mobile phone to demonstrate digital technologies and ask learners what they use it for – emphasising responsible use such as safe use, etc.	
Assessment – Term 1	
Continuous Assessment – Refer to Section 4	

3.1.2 Term 2

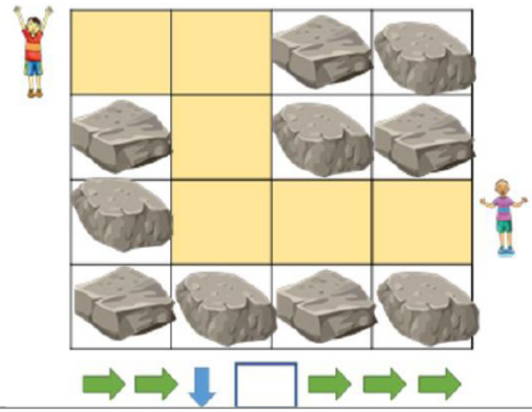
Content (Grade R / Term 2)	Notes/Examples
Pattern Recognition (± 2 hours)	**Could be integrated with Mathematics**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Reinforce from Term 1**
Identify a pattern: Identify a complete pattern presented as a data set (reinforce from Term 1 using different and more complex examples/ activities) **Recognise pattern:** Recognise and interpret patterns in symbolic sets of data or visualisations **Example: Recognise a pattern:**   What type of rock needs to be placed on the X to complete the pattern?	Identify a pattern is reinforced from Term 1 as it is used in recognising a pattern. Patterns take various forms and are found in poems, music, dance, symbols, etc.
Algorithm Design and Coding	**Algorithm Design and Coding mostly go together.**
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem	**Reinforced from Term 1.**
Examples: Sequence the two sets of pictures.  DBE Grade R Workbook 1 (English) – p41  What is the missing command to be paced in the ☐ ?	It is further reinforced when C.2 is introduced as it is also used in Term 2 with C.2. Before a solution / set of instructions is presented, it is designed and developed using computational thinking. (One arrow represents movement until barrier is hit) In Term 2, C.1 could be used with C.2 **Note:** For Grade R and Grade 1, the directional arrows imply/include a change of direction of the sprite/character to automatically face the direction for following the path in which it must continue. For terms 1 and 2 of Grade R the solutions could be presented in simplistic form, i.e. an arrow represents movement forward till it hits the barrier (no matter how many steps/blocks).

Content (Grade R / Term 2)	Notes/Examples
Simplistic movement solution (move till barrier is hit) **vs.** **Consice step movement** (move with single seps / one block at a time)   	From term 3 onwards, each arrow can represent a single step/movement, i.e. on block at a time. **Note:** ***Coding*** is the process of creating a logical set of instructions that a human or computing device can understand and execute, which require a deep understanding of computational thinking and problem solving
C.2 Present a simple coding solution using symbolic or written statements that represent sequences of commands, single repetition and conditional constructs.	**Links to C.1 and C.3.**
Example 1 Simplistic coding problem with solution provided and with ***Start*** and ***Stop*** events.  **Example 2** Simple coding problem without solution  	Sometimes an activity will combine competencies, e.g. combine C.2 and C.3 In the first example, coding instructions are presented, and learners need to interpret and execute the instructions (C.3). In the second example, learners must first identify and choose the correct arrows, then they can pack the arrows in the correct order (develop a set of logical instructions (C.1) and present the solution (C.2) and eventually follow/execute the instructions or acting out their solution (the code) - (C.3). In some problems, the combination of competencies happens naturally and intuitively and cannot always be separated. The more complex problems grow, the more competencies are included.

Content (Grade R / Term 2)	Notes/Examples
C.3 Interpret and execute a given symbolic or written set of commands	**Links to C.2. Can also link to R.6**
Example 1: The butterfly must reach the flower. Restriction of an instruction set with only 4 instructions.  **Example 2:** Rules for crossing the street. 	Reinforce from Term 1 Provide learners with a grid scenario and symbolic commands or a real-life scenario with instructions to follow when crossing the street, which they must interpret. Initially the grid size should not be larger than 3x3
Robotics	**Could be integrated with Creative Arts**
R.1 Explain what a robot is in simple terms	**Reinforce from Term 1**
R.2 Identify different types of robots **Examples:** Provide examples of robots and non-robots and let learners identify the robots. Then identify the type of robots and discuss the different tasks it performs.	Use as prior knowledge of what a robot is to identify types of robots. Can be done with R.1
R.5 Design a simple product (artefact) based on a set of design specifications	
Example 1: Design a robot  **Example 2: Make a wiggly worm**  https://www.youtube.com/watch?v=xMAPpjdWQq8	Provide instructions/specifications (visually or verbally) Bring in idea of movement (robot can 'do' something) The learners could be given an empty template of a robot and asked to fill the robot with patterns. The appearance of the robot could be enhanced by adding googly eyes. Small motor skills are also developed if the learners are tasked to cut certain patterns to complete their robot. The arms, legs and head of the robot can then be attached to small pieces of pipe cleaner. This will allow the parts to be moved. The learners can then in a separate activity be tasked to code simple dance moves for the robot.
R.6 Mimic the operations of a robot	**Links to coding; set of instructions (algorithm)**

Content (Grade R / Term 2)	Notes/Examples
Reinforce from Term 1 using different and more complex activities. **Example:**  	Reinforce from Term 1 using different activities and gradually increase complexity of activities. Carry out a pattern of robotic movements. Teacher could highlight aspects about what a robot is such as: • Robot has arms and legs. • Robot follows instructions. • Robot can move according to instructions. Also link to C.2 and C.3
Digital Concepts	**Could be integrated with Life Skills**
D.2 Recognise that he or she is living as citizens in a digital world	**Reinforce and extend from Term 1**
Reinforce and extend from Term 1, e.g. Protecting personal information: Explain concepts, using examples/pictures/role play relevant to their daily lives, such as: What is personal information, Why is it important to keep it private?	D.2 and D.3 can be done together. Reinforce and extend from previous terms with different examples and activities. Aspects that could be discussed: • Limited screentime • Protection of personal information
D.3 Demonstrate an understanding of the concept of a computing device	
Reinforce from Term 1. Learners can bring pictures of devices and explain how the device receives input, what input It receives and how it provides output and what output it provides. How does the device know what output to provide?	
Assessment – Term 2	
Continuous Assessment – Refer to Section 4	

3.1.3 Term 3

Content (Grade R / Term 3)	
Pattern Recognition:	**Could be integrated with Mathematics**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Links to C.1, C.2 and C.3**
Identify a pattern: Identify a complete pattern presented as a data set (repeat from Term 1&2 using different examples/activities) **Recognise pattern:** Recognise and interpret patterns in symbolic sets of data or visualisations (repeat from Term 2 using different examples/activities) **Copy:** Copy a pattern presented as a dataset **Example:** Identify, recognise, interpret and copy (mimic) the following pattern, then complete the pattern Patterns can have various forms: Poems, music, dance, symbols, etc.  DBE Grade R Workbook (English) – p15 The example above incorporates words and repetition. Although this is verbal and symbolic for Grade R, it could be adapted for later grades to include written text.	Reinforce and extend from Term 1 Patterns take various forms and are found in poems, music, dance, symbols, data, etc. This could be integrated with Languages as well. Pattern recognition is the process to identify and extract meaningful patterns from a dataset. It involves using analysing a set of data to find regularities or repeating structures that can be used to make predictions, classify objects, or solve problems.
Algorithm Design and Coding	**Algorithm Design and Coding mostly go together.**
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Links to C.2**
Example: What is the missing command to be place in the in the blue square? **Note**: The first arrow is to step onto the grid and the last arrow is to step out of the grid. This type of scenario is case-dependent – other times they will start from within the grid and end inside the grid. 	In C.1 a partial solution is given, and the learner must complete the solution. Providing partial solutions serves as scaffolding for developing the complete solution. C.1 and C.2 is continuously reinforced from previous terms as computational thinking is always used to design and develop an algorithm/solution before presenting the solution. Once the set of instructions (solution) is developed, it can be translated to code: in this instance arrows

Content (Grade R / Term 3)	
C.2 Present a simple coding solution using symbolic or written statements that represent sequences of commands, single repetition, and conditional constructs	**Links to C.1 and C.3**

Translate (code) the solution (algorithm) into arrows (code) – this is the solution to the problem (Links to C.1)

C.3 Interpret and execute a given symbolic or written set of commands	**Links to C.1, C.2 and C.4**

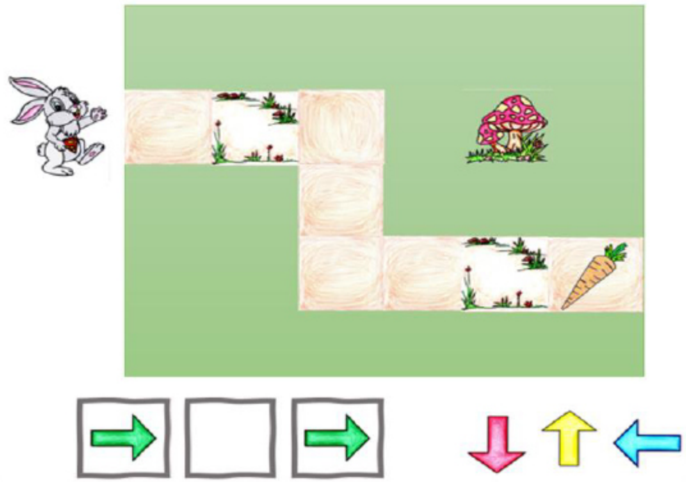
Example 1: Interpret code

Interpret the code (from the above solution), then execute the code

Once the set of instructions is developed and translated into code (symbolic code (arrows), another learner can interpret it and act it out on a grid.

Example 2: Complete code

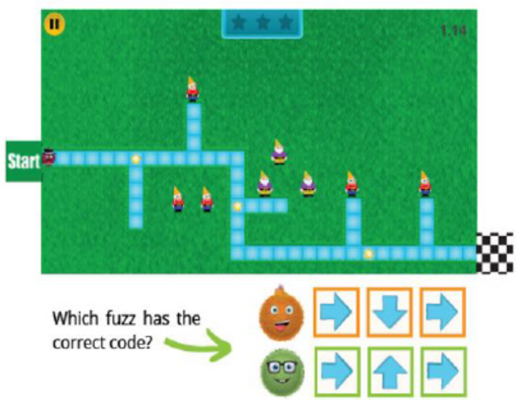
Provide the missing instruction for SSB to follow the path and get to the carrot.



Provide various activities to develop interpretation and execution of code:

- Provide code for learners to complete.
- Provide code that learners need to mimic/act out.
- Provide code, leaving out a step that learners need to fill in.

C.4 Debug a given symbolic or written set of instructions.	**C.1, C.2, C.3 and C.4 are all steps in algorithm design.**



Find the error in the given instructions
https://www.kodable.com/hour-of-code/unplugged-coding

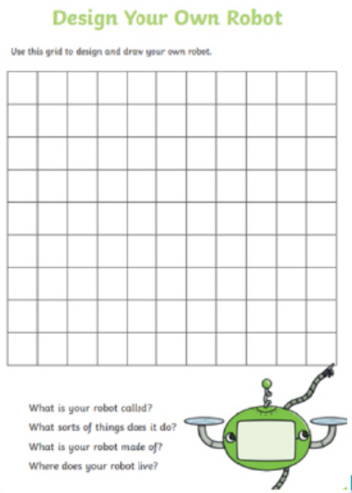
Find the 'errors' in the given picture

Find the error in the given pattern. The learners can also be given an option to correct the pattern with the necessary object.

Once the set of instructions are acted out on a grid, it is possible to see if it works correctly.
If it does not work correctly, the learners must find the problem and correct it.

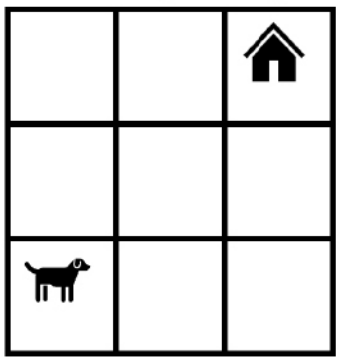
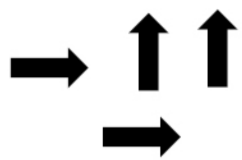
Sometimes, activities with incorrect solutions should be given to learners and they need to find the problem and correct it as is the case with the examples on the left.

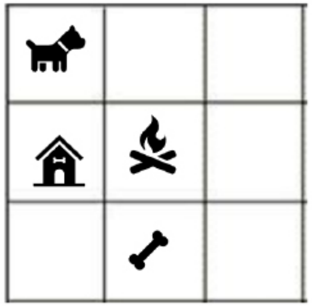
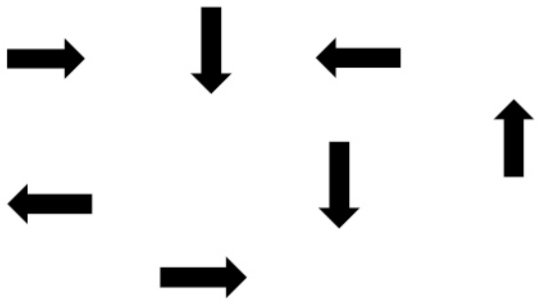
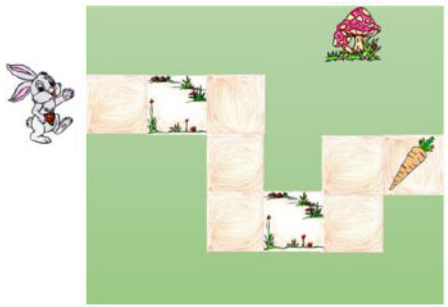
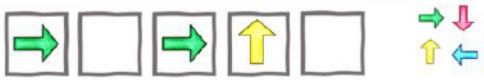
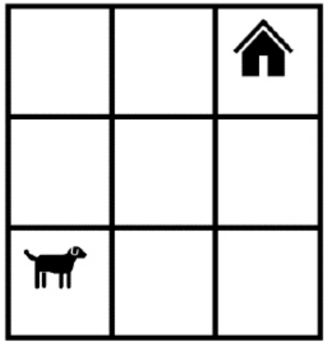
https://www.kodable.com/hour-of-code/unplugged-coding

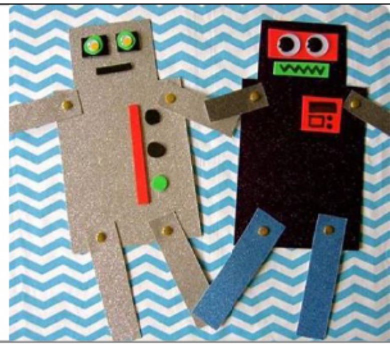
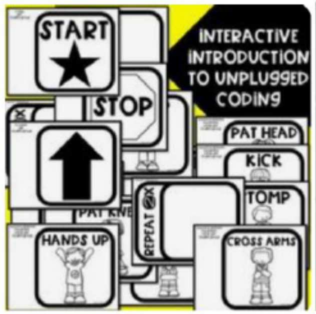
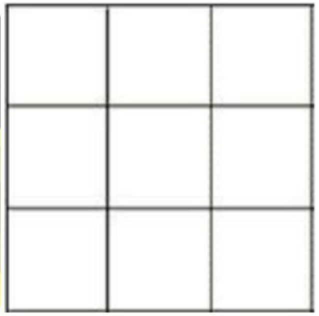
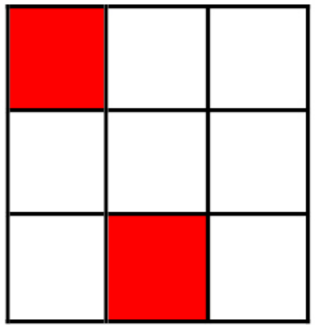
Content (Grade R / Term 3)	
Robotics	**Could be integrated with Creative Arts and Language**
R.1 Explain what a robot is in simple terms.	**Reinforce and extend from previous terms with different examples and activities**
R.2 Identify different types of robots.	
R.5 Design a simple product (artefact) based on a set of design specifications.	**Could be integrated with Life Skills (Creative Arts)**
Example: Design you own Robot  Robots Activity Booklet (Ages 5 - 7) (teacher made) - Twinkl After designing their robot, learners answer questions such as: • What is your robot called? • What things can your robot do? • What is your robot made of? • Where does your robot live?	Learners can ideate (design thinking development) about their robots.
R.6 Mimic the operations of a robot.	**Links to C.3**
Example: Using card blocks with arrows and picture instructions, learners mimic a robot carrying out instructions 	Robots only act upon instructions (only follow instructions). The learner can act as a robot and act out the instructions provided to the robot. Instructions could be in symbolic or written format or verbal.
Digital Concepts	**Could be integrated with Life Skills and Language**

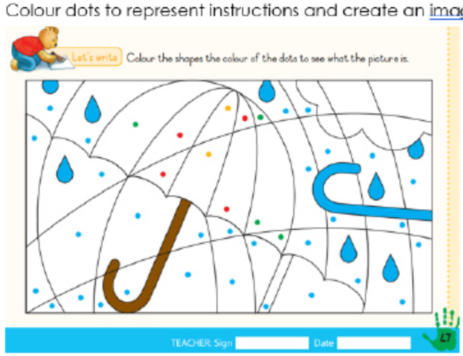
Content (Grade R / Term 3)	
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.3**
Possible discussion: **Limit screen time** Need to balance time spent on screens with other activities such as playing outside, spending time with family and friends, listening to stories (e.g. someone reading to them), etc. **Protecting personal information** What is personal information? Why is it important to keep it private?	Reinforce and extend from previous terms with different examples and activities. • Limited screentime • Protection of personal information
D.3 Demonstrate an understanding of the concept of a computing device.	**D.3 can be done in relation to D.7**
Example A computing device is a machine that can receive input, do something with the input and provide a result or output, for example, a mobile phone can be used to play games or watch videos that is stored on the phone.	Reinforce and extend from previous terms. A computing device receives input, that results in output after processing.
D.7 Present a basic understanding of the concept of input processing and output.	**Link to D.3**
Example Using a mobile phone, illustrate the concepts of input and output: a touch screen serves as both input and output device. In order to get output, the user must provide input, e.g. pressing the phone icon. Demonstrate, using a mobile phone or remote-control toy, how input results in output.	Reinforce and extend from previous terms with different examples and activities Input triggers instructions that results loop in output. Output could be on the screen or through earbuds when listening to music.
Assessment – Term 3	
Continuous Assessment – Refer to Section 4	

3.1.4 Term 4

Content (Grade R / Term 4)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Reinforce and extend from previous terms, gradually increasing complexity.**
Identify a pattern: Identify a complete pattern presented as a data set (repeat from Term 1 using different examples/activities) **Recognise pattern:** Recognise and interpret patterns in symbolic sets of data or visualisations. **Copy:** Copy a pattern presented as a dataset **Complete:** Complete a pattern presented as a dataset **Example:** Complete the pattern 	
Algorithm Design and Coding	**Algorithm Design and Coding mostly go together.**
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Reinforce and extend from previous terms using different activities that gradually increase in complexity**
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, repetition and conditional constructs.	
C.3 Interpret and execute a given symbolic or written set of commands	
Example 1: Pictures communication for washing hands  **Example 2:** Use the arrow cards provided and pack them in the correct sequence so that the dog can return to his home. Then act as the dog and follow your instructions on the grid.   **Note**: The above problem has more than one solution (route) Ask learners how many routes can be packed using the available cards. Let learners compare their solutions and discuss their different routes. **Example 3:** Pack the cards provided and put them in the correct sequence so that the dog can first pick up its bone and then return to its home.	First example: learners need to sequence pictures in the correct order. Second example: learners need to interpret the instructions provided and pack them on the grid to get the dog home. **Note**: Example 2 is an example of a Parsons puzzle type of problem. In terms of problems that provide a partial solution where some code instructions are missing and learners must fill in the missing code instructions, the concept of Parsons Puzzles could be helpful as it provides scaffolding for learning programming. It helps learners to develop logical thinking, The concept is a type of scaffolded program construction tasks where the learner is given a set of code blocks of a single or multiple lines of code, and the task is to piece together a program from these or to fill in missing code from these. Parsons programming puzzles are an evidence-based teaching practice that reduces the cognitive load and time spent for learners (Parsons & Haden, 2006).

Content (Grade R / Term 4)	Notes/Examples
   **Example 4: Provide missing code** Follow the instructions and provide missing code from the code instructions provided Sipho Super Bunny (SSB) need to collect the carrot. Fill in the missing instructions from the code provided in the right:  	Third example: Now, the complexity can be increased by adding obstacles (fire) and limitations/conditions (must first pick up the bone). The learner must interpret the grid and the instructions provided, then pack the arrows in the correct sequence to solve the problem. The learner now must use C.1, C.2 and C.3 in combination to complete the task, which also increases complexity. Can also link to R.7 **Note:** Complexity of problems can also be increased by • providing more arrows than required and/or • a bigger grid with limitations that rule out obvious solutions and/or • Provide code with missing instructions
C.4 Debug a given symbolic or written set of instructions.	**Link to C.1, C.2 and C.3**
Example: 1 Tumelo used the following instructions to solve the problem on the right.   Act it out the above instructions to see if it is correct. If it is not correct, find the error and correct it. **Example 2 (pair programming)** A learner/group of learners are provided with a grid with obstacles and a set of instructions to meet an outcome. One learner acts as a robot and execute (act out) the instructions developed to see if it works. If there is a mistake, the next learner/group of learners must debug and correct, then test again…repeat until it is correct	Provide an incorrect set of instructions to learners which they need to act it out to determine if the solution is correct, find the error and correct it.

Content (Grade R / Term 4)	Notes/Examples
C.5 Evaluate a given solution towards potential improvement.	**Links to C.1, C.2, C.3 and C.4**
Example 1 Provide learners with two possible solutions/algorithms/set of instructions, e.g. routes on a grid that achieve the same outcome. Let them act both out and decide which one is the best /most efficient (shortest) solution. **Example 2** Provide learners with a set of instructions to find an item on a grid that could be shortened (not the shortest route to the item). Ask them to determine if there is a possible shorter route, and provide the shorter route instructions	Assess 2-3 solutions provided for effectiveness and determine which is the most effective and Assess a solution/set of instructions to see if it could be improved, e.g. shortened.
Robotics	**Could be integrated with Creative Arts and Language**
R.1 Explain what a robot is in simple terms.	**Reinforce and extend from previous terms with different examples and activities**
R.2 Identify different types of robots.	
R.4 Present an understanding of how robots affect the world.	**Link to R.1 and R.2**
Example Ask learners how they think robots can help or harm us. Discuss possibilities such as robots doing work in dangerous places where humans cannot go, robots used in industry for assembling cars, etc.	Robots are changing the world by helping humans do things with greater efficiency and doing things that were not possible before
R.5 Design a simple product (artefact) based on a set of design specifications.	**Can link to R.4**
Example activity: Robot with cardboard and paper pins /split pins with moving arms and legs. 	Learners should ideate about their robots: who they are and what they can do, how they can help people. Linking to R.4, let learners explain how their robot will affect the world, answering questions about their robots (the explanation can be integrated with Language). Adding string to make the parts move, the learners can write symbolic code using coding cards for the robot to perform instructions. Such as, move right arm, move left arm etc.
R.6 Mimic the operations of a robot.	**Link to C.1, C.2, C.3**
Example activity Use instruction cards with pictures/images on a grid to lay out instructions for a robot that learners can act out. Incorporate limitations such as obstacles/no-go areas, etc.  	Adding limitations and obstacles increases the complexity of the task, e.g. *no go* areas such as red blocks indicating hot lava 

Content (Grade R / Term 4)	Notes/Examples
R.7 Create test and execute a set of robotic instructions.	**Links to C.1, C.4 and C.5**
Example: Provide a grid with a problem to be solved using coding cards. • Learners solve the problem by developing the algorithm and coding it using coding cards. • Learners then execute (implement) their solution on the grid to test it. • Leaners then correct their solution/instructions if not correct	Reinforce and extend using different activities– see previous examples.
Digital Concepts	**Could be integrated with Life Skills and Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Link to D.2, D.3, D.7**
Examples of technology include computers, smartphones, TVs, video games and even 'robots' that perform specific tasks. Information Technology (IT) is a type of technology that deals with information, such as data, images, and sound. IT includes things like computers, software, and the internet. The **purpose** of IT is to help people access and use information more easily and efficiently	Technology is all around us and we use it every day to communicate, learn and have fun. Provide examples of technology
D.2 Recognise that he or she is living as citizens in a digital world.	**D.2 and D.3 can be done together.**
Limit screen time Need to balance time spent on screens with other activities such as playing outside, spending time with family and friends, listening to stories (e.g. someone reading to them) **Protecting personal information** What is personal information? Why is it important to keep it private?	Reinforce and extend from previous terms with different examples and activities. • Limited screentime • Protection of personal information Teaching young children about the impact and dangers and benefits of being a digital citizen is an ongoing process that requires regular reinforcement using various examples
D.3 Demonstrate an understanding of the concept of a computing device.	**D.3 can be done in relation to D.7**
Reinforce from previous terms Input triggers instructions that result in output – demonstrate using various types of output devices (screen, microphone/earbuds) for different types of output such as text, sound	Reinforce and extend from previous terms with different examples and activities.
D.7 Present a basic understanding of the concept of input processing and output.	**Revise and reinforce from previous grades using different examples/devices/pictures and activities.**
D.8 Interpret a pattern to represent or communicate a message or image.	**Links to C.6**
Example The image below provides a message about tomorrow's weather. Use the colour codes to colour the blocks/shapes and interpret the message provided after colouring. Colour dots to represent instructions and create an image  DBE Grade R Workbook 2 (English) – p47	

Content (Grade R / Term 4)	Notes/Examples
D.9 Create a pattern to represent or communicate a message or image.	**Done in relation to C.6 and D.8**
Example: Use a pattern to communicate, e.g. learner showing a blue circle with a red square to indicate that he/she is thirsty and needs to go to the bathroom. Use hand symbols:  	Learners could use symbols to communicate to teacher in class
Assessment – Term 4	
Continuous Assessment – Refer to Section 4	

Note:

In terms of coding, typically, problems could require learners to

- work through (trace) / act out code (execute) to determine the output or correctness, followed by debugging the code if necessary
- provide missing code instructions (code instructions are provided with some instructions or code elements intentionally omitted)
- choose the correct solution from 2-3 coding options
- compare different solutions to evaluate their efficiency
- translate verbal/written instructions into code (e.g. from instruction to packing arrows)
- rewrite a set of coding instructions to be more efficient, e.g. indicating number of times an instruction should be repeated instead of presenting each step sequentially or
- develop the solution algorithm (code instructions) themselves through the stages of planning, testing and debugging using computational thinking.

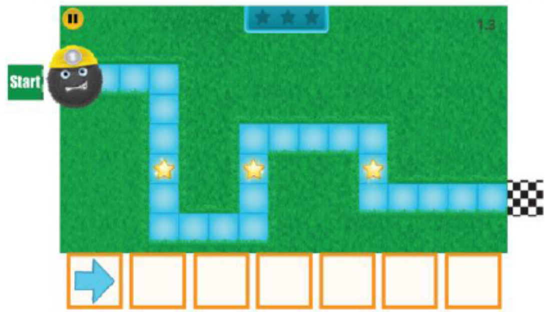
depending on the competency the learner needs to demonstrate

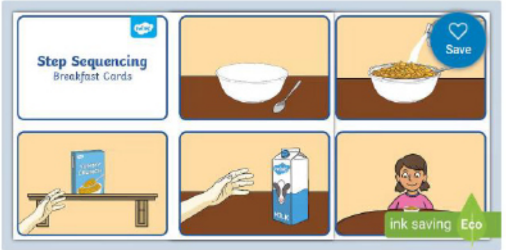
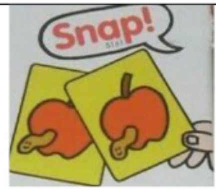
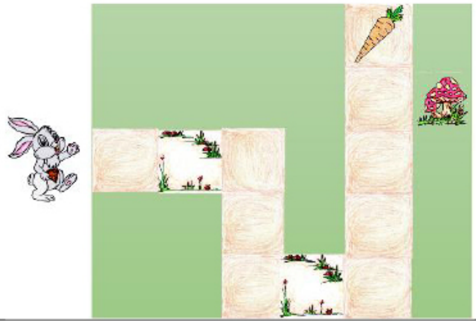
3.2 Grade 1

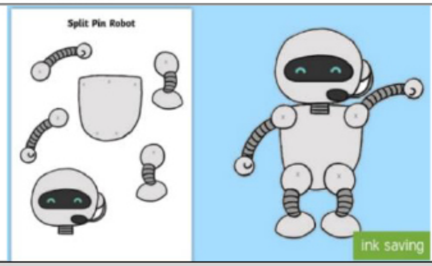
Note:

Teachers must include the following competencies and content in their Annual Teaching Plan (ATP), distributed across the terms and sequenced, organised and grouped (and integrated with other subjects in Grade 1 where applicable) in a manner that will facilitate learning ensuring ample retrieval and deliberate practise with feedback to ground principles and concepts, maximize the learners' learning outcomes and achievement but also ensure a gradual learning curve, and in a way that will make optimal use of time and resources.

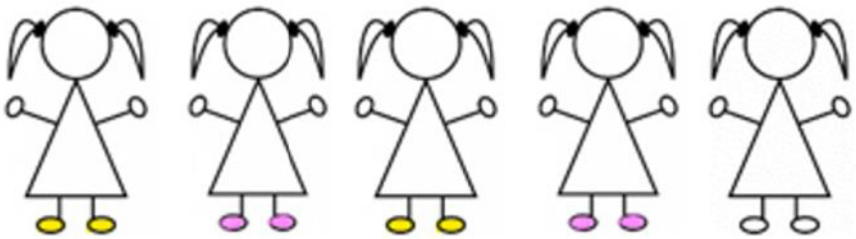
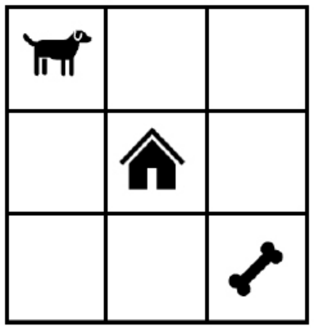
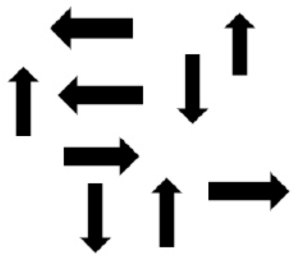
3.2.1 Term 1

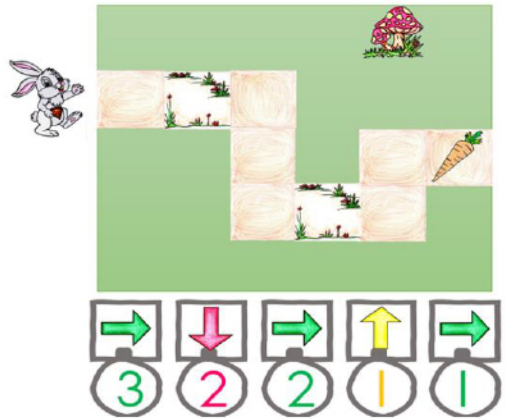
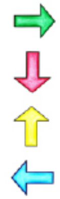
Content (Grade 1 / Term 1)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Reinforce and extend from Grade R**
Learners can recognise and interpret patterns with shapes, colours, or both.  Or numbers, e.g. 1 2 3 _ _ _ Or objects 	C.6 develops **pattern recognition** (part of computational thinking) that is eventually used to develop coding solutions as part of computational thinking to identify patterns in the coding problem and/or data by identifying similarities or differences that can help to solve the problem or refine the algorithm.
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**C.1 and C.2 can be done together**
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands using single repetition and conditional constructs.	Reinforce and extend from Grade R **Note:** Mostly, for Grade 1, like Grade R, the directional arrows imply/include a change of direction of the sprite/character automatically facing the direction to follow the path in which it must continue. **Note**: A coding solution (program) is a sequence of symbols (instructions) that specifies a specific task to be completed
Example 1 Draw the missing arrows to show the fuzz how to get through the maze  Note: Move until barrier is hit (not one block at a time) https://www.kodable.com/hour-of-code/unplugged-coding **Example 2 – Picture Story** Retell a story with 3 to 5 pictures that learners put in sequence.	

Content (Grade 1 / Term 1)	Notes/Examples
 https://www.twinkl.co.za/resource/t-s-1640-7-step-sequencing-cards-eating-breakfast	
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.1 and C.2**
Example 1: Game-based rules and commands such as the game of match two cards or Snap can be presented and conceptualised.  **Example 2:** Follow the instructions and provide missing code from the code instructions provided Sipho Super Bunny (SSB) need to collect the carrot. Fill in the missing instructions from the code provided in the right: **Example 3 – Provide the missing code** Sipho Super Bunny (SSB) needs to collect the carrot. The following code is provided for SSB to get to the carrot.   Study the grid on the right and provide the missing code instruction for SSB to collect the carrot,	Reinforce and extend using different and more complex activities **Note** To ensure an acceptable learning curve/ progression, teachers can initially provide code that learners can choose from, e.g. provide learners with the following to code instructions to choose from to fill in the missing code, e.g.  As learners progress, they could be expected to fill in missing instructions without giving them instructions to choose from.
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	**Link to R.2**
Example: Explanations could include aspects such as: • A robot follows instructions / can be 'programmed'. • A robot has different parts. • Robots can perform actions according to instructions.	Reinforce from Grade R Learners explain what a robot is in their own words, according to their own understanding. Teacher can ask questions to elicit answers.
R.2 Identify different types of robots.	**Link to R.1**
Reinforce from Grade R using examples such as: • Domestic, e.g. robot vacuum cleaners • Industrial, e.g. robots that assemble cars • Education, e.g. to learn to code such as programmable robots like LEGO Mindstorms and Dash and Dot Learners should understand that Robots can come in all shapes and sizes, from tiny ones that clean floors to giant ones that build cars! They have different parts, like a brain (called a computer), eyes (called sensors), and arms and legs (called actuators) that help them move and work	Reinforce and extend from Grade R Provide pictures of different types of robots and discuss the 'work' that they do.

Content (Grade 1 / Term 1)	Notes/Examples
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.1 and R.2**
Example: Using paper pins / split pins and a design template 	Learners ideate about their robots. Who they are, what type they are and what they can do, how they can help people. (the ideation can be integrated with Language). One could add strings, e.g. to the arms to make the arms move and learners could write symbolic code using coding cards for the robot to perform instructions.
Digital Concepts	**Could be integrated with Life Skills or Language**
D.2 Recognise that he or she is living as citizens in a digital world. We live in a technology-driven world with computing devices all around us. **Example**: We need to: **Limit screen time** Need to balance time spent on screens with other activities such as playing outside, spending time with family and friends, listening to stories (e.g. someone reading to them or start reading themselves as soon as they can), **Protecting personal information** What is personal information, Why is it important to keep it private?	Reinforce and extend from Grade R with different examples and activities. -limit screen time -protect our personal information
D.3 Demonstrate an understanding of the concept of a computing device.	**Links to D.2**
Example Use learners' pre-knowledge about the concept of input-processing-output to explain, in simple terms, what a computing device is. **Possible explanation** A computing device is something you can use to do things like play games, watch videos, or talk to people who are far away. Computing devices have screens where you can see what you are doing, and you can use buttons on touch screens to control the device They also have something called a processor that helps them to do the things we want them to do.	Extend from Grade R A computing device is a machine that can work with information such as numbers, words, pictures, movies or sound. This information is also called data. It can also store information/ data such as a phone number or photos and can display the information or data if you give the correct instructions. A computing device can also do computations and can process data very quickly.
Assessment – Term 1	
Continuous Assessment – Refer to Section 4	

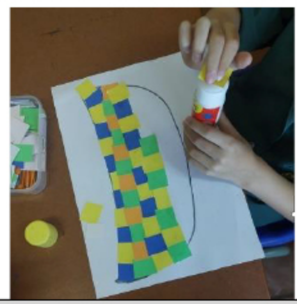
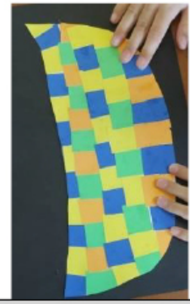
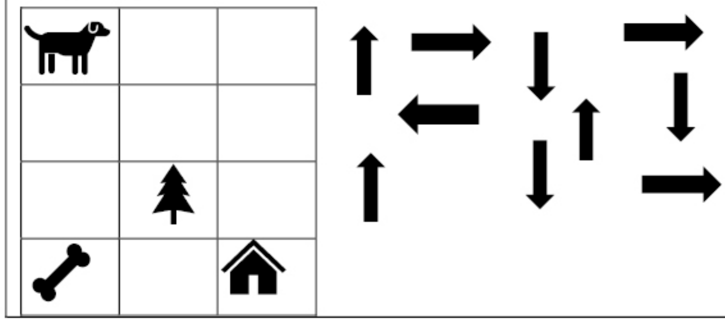
3.2.2 Term 2

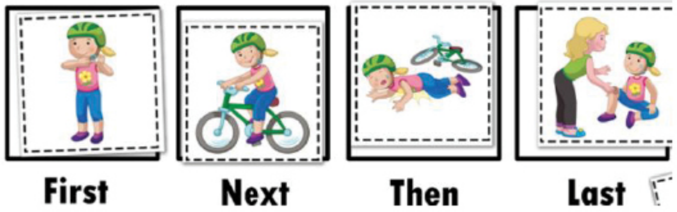
Content (Grade 1 / Term 2)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Link to C.1**
Example 1: Complete the pattern **5 6 7 _ _ 10** **Example 2:** What colour/s will the last doll's shoes be?  Example 1: Pattern rule: +1 (count in ones). Example 2: Pattern rule: yellow, pink, yellow, pink… Stick Figure Kids Clip Art - Cliparts.co **Example 3** What are the steps you follow to buy things in a store, e.g. select products, scan at till, pack in bag, pay cash or with card. Learners can communicate their pattern or make use of pictures to demonstrate the sequence of events, e.g. provide a selection of pictures in random order demonstrating a person with a trolley, a cashier scanning items, a person paying at the till, items being packed in bags	By identifying patterns, we can predict what will comes next and what will happen again and again in the same way. A pattern may be numerical, visual or behavioural. Pattern recognition eventually leads to analysing patterns in data.
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**C.1 and C.2 can be done together**
Example: Learners develop a solution using computational thinking. Provide learners with directional cards. Learners develop a set of instructions for the dog to pick up the bone and go home, using the cards received.   **Note** From the arrows provided, this problem could have more than one solution. Let learners evaluate and discuss their solutions.	Learners create their own solutions using code cards (arrows, letters, words) which they present. **Note**: A problem such as this example could be adapted to ensure that there is only one solution or when more than one solution, one route may be more effective (shorter), asking learners to evaluate the effectiveness (shortest route) and recommend the most effective route, etc.
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.	**C.2 and C.3 can be done together**
	The concept of repetition coding construct (loop) is introduced using numbers below the arrows.

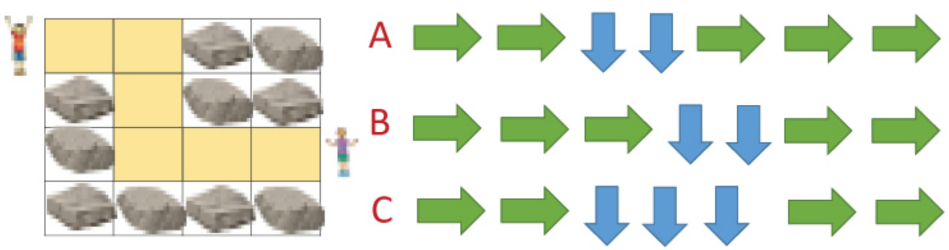
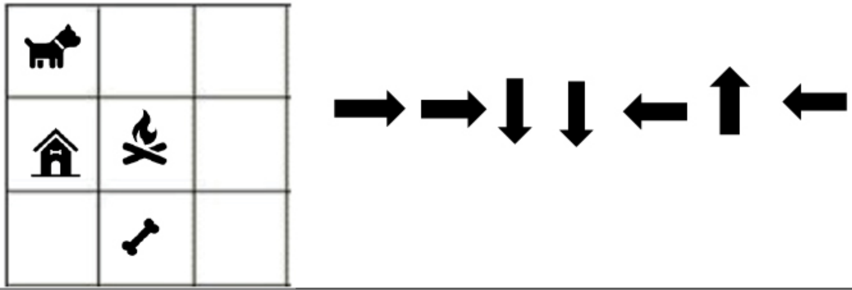
Content (Grade 1 / Term 2)	Notes/Examples
C.3 Interpret and execute a given symbolic or written set of commands	The number below the arrow indicates the number of times the instruction must be repeated, e.g.  has the same meaning as 
Example 1: Sipho Super Bunny (SSB) can only move one block at a time. Study the image below and write down the instructions SSB must follow and how many times SSB must carry out each instruction to get to the carrot.  **Example 2:** SSB must walk along the path and collect all the carrots. Study the following image and answer the questions that follow.  • How many carrots have SSB collected when it reaches the mushroom? • How many carrots will SSB have collected when he reached the end of the path?	The learners in grade 1 can also substitute the arrow symbols with characters representing various actions.  The solution for Example 1 can therefore also be represented as: 
Robotics (± 2 hours)	**Could be integrated with Creative Arts**
R.1 Explain what a robot is in simple terms.	R.1, R.2 and R.3 can be done together First learners explain what a robot is and name different types of robots which can give them inspiration for designing a robot (R.5). Then, provide instructions to design/draw a robot Then they can ideate about the type of their robot, what it can do and how it will impact the world
R.2 Identify different types of robots.	
R.5 Design a simple product (artefact) based on a set of design specifications.	
Example: Provide design instructions for designing/drawing the robot	
R.6 Mimic the operations of a robot.	**This happens in relation to C.1, C.2, C.3 and C.4**
Some activities could be reused and presented in a game board format. In the example below coding cards are printed and laminated and SSB is held up with a paper binder. The learners can each be presented a set of coding cards. The different laminated mazes can be exchanged. The coding cards and icons are reusable. Some coding cards may be left blank where instructions could be added and wiped off. 	Learners use directional coding cards to pack out the instructions and then act out the coding instructions.

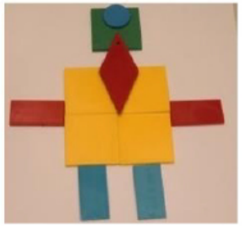
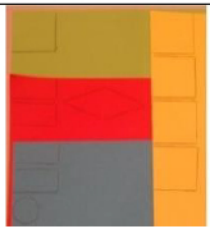
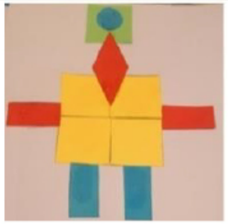
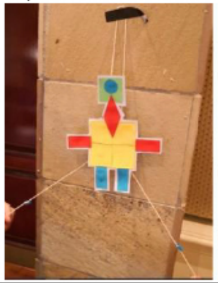
Content (Grade 1 / Term 2)	Notes/Examples
Digital Concepts	**Could be integrated with Life Skills and/or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Reinforce and extend**
Technology is designed with a purpose of solving problems that meet human needs and wants. It refers to tools, machines, or devices that make our lives easier or better. **Examples** of technology include computers, smartphones, TVs, video games and even 'robots' that perform specific tasks. Information Technology (IT) is a type of technology that deals with information, such as data, images, and sound. IT includes things like computers, software, and the internet. The **purpose** of IT is to help people access and use information more easily and efficiently.	Reinforce from Grade R using different examples and activities. Technology is all around us, and we use it every day to communicate, learn, and have fun.
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1**
Interact appropriately with others online. **Using appropriate examples/role play, etc. discuss:** What does it mean to interact online? Why is it important to treat other with kindness and respect? Emphasise safe practices such as not sharing personal information and limiting screen time.	Reinforce from previous terms: Limit Screen time and Protect Personal Information Extend to: Interact appropriately with others when online
D.3 Demonstrate an understanding of the concept of a computing device.	**Reinforce and extend from Grade R**
Provide examples and pictures of computing devices and explain what makes these devices computing devices. **Example**: A computing device is a machine that can receive input, do something with the input and provide a result or output, for example, a mobile phone can be used to play games or watch videos that is stored on the phone. If you provide the correct instructions, it can open the video so you can watch it. It can also do some mathematics such as calculations.	Reinforce and extend from previous grades and terms using different examples and activities
Assessment – Term 2	
Continuous Assessment – Refer to Section 4	

3.2.3 Term 3

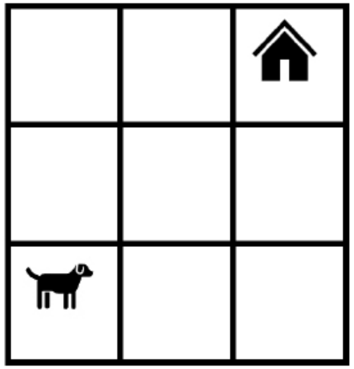
Content (Grade 1 / Term 3)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Reinforce and extend from previous terms**
Example: What colour/s will the last doll's shoes be?  The first doll has two yellow shoes. The second doll has one yellow and one pink shoe. The third doll will then have two pink shoes. If the fourth doll has one pink and one purple shoe, the fifth doll will have two purple shoes. Stick Figure Kids Clip Art - Cliparts.co	Identify the pattern rule to predict the next doll's shoe colours. Adapt activities to suit learners' level.
C.7 Create or complete a pattern to represent a data set.	**Link to C.6**
Example 1: Brushing your teeth: Give learners a set of instructions consisting of pictures. Leave out important steps, e.g. taking the cap off the toothpaste tube. They must pack the activity in sequence and then test and debug. **Example 2:** Learners follow instructions/patterns to design their own fruit basket pattern, by sticking and cutting coloured paper in alternate pattern formations. The final product is then cut out and presented. The completed pattern can then be used as part of a larger design.  	Learners to order into correct sequence and complete the sequence. They can only test if they execute the code (do it themselves).
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Link to C.2**
Example: Develop instructions for the dog to first pick up the bone and then go home. Dog starts walking in the direction it is facing and must avoid the tree.  **Note**: Though there is more than one possible route, the directional cards provided limit the possible solution(s)	Reinforce and extend from previous terms using different activities that gradually increase in complexity, e.g. coding a problem on a grid with obstacles or limitations. **Note**: This problem can be adapted to provide more directional cards (or none – open ended) and then ask learners to find the shortest route or to find all the routes, compare the different routes and identify the shortest route, etc.

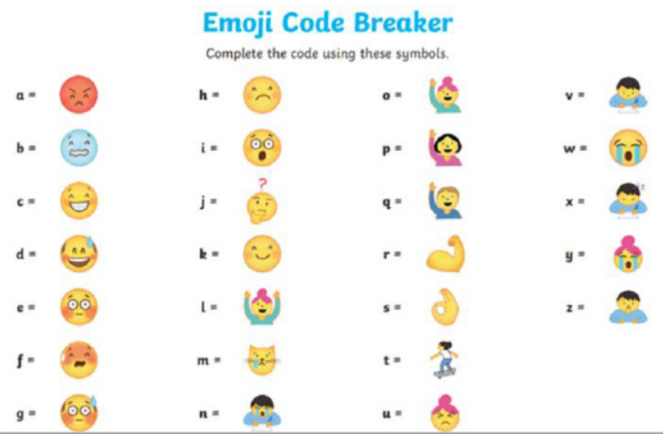
Content (Grade 1 / Term 3)	Notes/Examples
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition and conditional constructs.	**Link to C.1** **Learners can act code developed in C.1.**
Example 1: Learners act out code (solution in C.1) to see if it is correct. **Example** 2 – Picture Story  Every sequence has a beginning, a middle and an end (just like a story). **Beginning** What is the first step? Where do I start? **Middle** What are the in-between steps? What is the order of these? **End** What is the final step? How does it end?	Picture stories are presented in a logical order (sequenced correctly with a beginning and an end). When ***sequencing***, we learn about patterns and relationships, and to understand the order of things. By learning to sequence, we develop the ability to understand and arrange purposeful patterns of actions, behaviours, ideas, or thoughts that supports the logical sequencing of coding instructions.
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.1 and C.2**
Example 1 **Sequencing & Problem solving** In the DBE Math English grade 1 workbook (p 69) the days of the week are given in sequence. Such an exercise can easily be converted to a sequencing and problem-solving activity. The example below can be phrased as a problem as follows: SSB must identify the days of the week in the correct order. How should SSB walk to cross each day but not cross the same tile twice?  **Example 2** Study the following image and answer the questions that follow.  - How may carrots will SSB have collected if he stands on block A? - How may carrots will SSB have collected if he stands on block B? - How may carrots will SSB have once he as collected all the carrots? - The carrots follow a certain pattern how many carrots must be placed on block C to complete the pattern?	Learners interpret and execute code / pretend to be robots following instructions **Note** Evidence suggests that pupils should be taught – initially at least – in small bite-sized chunks. These steps in the learning process should be well-thought out and gradual as well as allow plenty of opportunity for practise (see, for example, Rosenshine, 2012; Coe et al., 2014; Sealy, 2019).

Content (Grade 1 / Term 3)	Notes/Examples
C.4 Debug a given symbolic or written set of instructions.	**Link to C.1, C.2 and C.3**
Which solution A, B or C is correct to have Ben meet Thatho?  **Note**: The child must first step onto the grid (using one arrow or stepping onto) and in the end step outside (using on arrow for stepping outside)	Learners evaluate and/or execute different solutions to find the correct solution. Note: the learner must first step
Example 2: The dog must pick up the bone and go home. Execute the code provided and debug if necessary. 	Test and debug: Leaners execute code, find the error, and correct the code
C.7 Recognise and interpret patterns in symbolic sets of data or visualisations.	
Create a code for each colour peg e.g. (B, R, Y). Write down the code set → B R Y B R Y R B Y Debug the code set → B R Y B R Y B R Y 	
Robotics (± 2 hours)	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	Reinforce and extend from previous Grades and Terms using different examples and activities. R.1, R.2 and R.3 can be done together using various examples of robots. Robots can have a body, arms and hands, sensors, a power source, wheels or legs and attachments.
R.2 Identify different types of robots.	
R.3 Outline the different components of a robot	
Example: Show learners pictures of robots and non-robots. Then they identify the robots and identify the different components of the robots. Learners explain in their own words what a robot is, according to their understanding, while the teacher prompt with *what*, *why* and *how* questions	

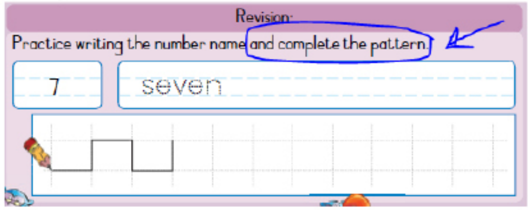
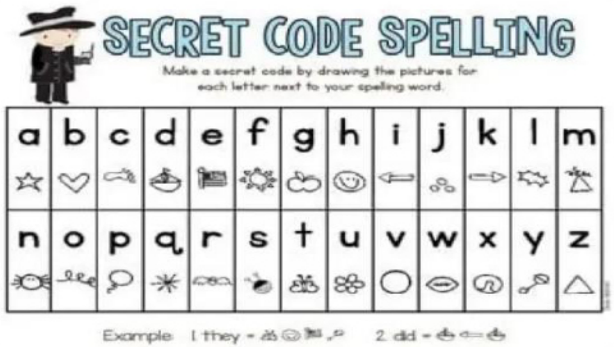
<table>
<tr><th>Content (Grade 1 / Term 3)</th><th>Notes/Examples</th></tr>
<tr><td>R.5 Design a simple product (artefact) based on a set of design specifications.</td><td>Link to R.1, R.2 and R.3</td></tr>
<tr><td>Example: Make a string pull robot.
<table>
<tr><th>Pack out robot with blocks</th><th>Trace blocks (parts that make up robot) onto paper/ cardboard</th><th>Cut out parts and glue onto white paper</th><th>Cut out robot</th><th>Turn robot face down.
Use two straw pieces and stick to back.
Use string to manipulate ‘robot’</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
</table>
Implement and test the design
 </td><td>This activity involves steps and incorporates several skills. Each learner first designs their own geometric robot using existing shapes (from a shape set) on paper. Then these are traced and cut out on coloured paper (developing fine motor skills) then stuck to a cardboard and cut out again. Two pieces of straw are attached with a string in a ∩ shape with stops added (two small pieces of straw)
Attaching the top to a wall and pulling the strings will result in the robot moving.

Learners ideate about their robots:
• What they can do
• How they can impact the world
• How they can be improved</td></tr>
<tr><td>R.6 Mimic the operations of a robot.</td><td>Link to R.7</td></tr>
<tr><td>Example:
Learner pretends that he or she is a robot that has been programmed to pick up and sort objects by colour and follow the following steps:
1. Pretend to turn on the power (e.g. Battery)
2. Use their sensors to "see" the objects that need to be sorted. We might have a camera or other device to detect colours.
3. Using their arms and hands, pick up one object at a time and examine it to see what colour it is. If it's blue, we would place it in the "blue" pile. If it's red, we would place it in the "red" pile, and so on
4. Repeat this process until all the objects have been sorted into their respective piles.
5. Once finished sorting, sensors are used to check and make sure that all the objects have been sorted correctly. If there are any errors, pick up the misplaced objects and move them to the correct pile.
6. Turn off the power to conserve energy for the next task</td><td>Learners act as robots and execute instructions</td></tr>
<tr><td>R.7 Create, test and execute a set of robotic instructions.</td><td>Link to R.6, C.1, C.2, C.4</td></tr>
<tr><td>Example 1:
Use the arrow cards provided and pack them in the correct sequence so that the dog can return to his home.
Then act as the dog and follow your instructions on the grid.
See how many alternative routes you can pack with the same cards.</td><td>Learners use the arrows provided to solve the problem.
Learners can compare their solutions to see if the problem could have more than one solution.
Challenge learners to find all the possible solutions.</td></tr>
</table>

Content (Grade 1 / Term 3)	Notes/Examples
  How many possible routes could be packed using the arrows provided?	Ask learners if any of the solutions is more efficient than the others.
Digital Concepts (± 1 hours)	**Could be integrated with Life Skills and/or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Link to D.2, D.3, D.4 and D.7**
Possible discussion: **Technology** Technology refers to the tools, techniques, and processes that are used to create, develop, and improve products, services, and processes. Technology is used to solve problems, improve efficiency, and enhance productivity. **Purpose of IT:** Information technology (IT) is a specific branch of technology that deals with the storage, processing, and transmission of digital data. The purpose of information technology is to provide tools and resources that enable people to manage and use information effectively	Technology is all around us. Provide pictures of examples of technology and information technology. Discuss the concepts of technology and IT as well as their purpose
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1**
Interact appropriately with others online. **Using appropriate examples/role play, etc. and discuss:** What does it mean to interact online? Why is it important to treat other with kindness and respect? Emphasise safe and good practices such as not sharing personal information and limiting screen time.	Extend and reinforce from previous terms. Interact appropriately with others when online
D.3 Demonstrate an understanding of the concept of a computing device.	**Link to D.7**
Possible discussion A computing device is something you can use to do things like play games, watch videos, or talk to people who are far away. Some examples of computing devices are computers, tablets, and smartphones. They have screens where you can see what you're doing, and you can use buttons or touchscreens to control them / give them instructions. It also has something called a processor that helps them do all the things you want them to do. So, when you use a computer or tablet, you're using a computing device.	Reinforce and extend from previous terms. Show pictures or examples of different computing devices to support understanding
D.4 Identify the common uses of ICT in the real world.	**Link to D.3**
Possible uses: ICTs refers to the tools and technologies that help us, process, share and communicate information. Common uses include communication, e.g. WhatsApp, entertainment such as video games, education using ICTs to learn, business, e.g. pay points, etc.	Use and discuss examples that learners are familiar with.

Content (Grade 1 / Term 3)	Notes/Examples
D.7 Present a basic understanding of the concept of input processing and output.	**Link to D.3**
Possible discussion: Input, processing, and output are the three basic components/processes of a computer system. **Input**: Refers to data or information that is entered into a computer system. For example, we use input devices such as a mouse, keyboard, microphone to enter the data or information. Input can take many forms such as text, images, voice. **Processing:** Processing refers to the manipulation of the data or information that has been input into the computer system. Processing transforms input into output. **Output**: Refers to the results or information that are produced by a computer system after processing the input data. This can take many forms, including text, images, video, and audio. Output is typically displayed on a computer monitor or screen, printed on paper, or played through speakers.	Provide examples or pictures of input and output devices to spark discussions. It is important that learners understand that these three components/processes work together to enable the computing device to perform a wide range of functions and tasks.
D.8 Interpret a pattern to represent or communicate a message or image.	**Link to C.6 and C.7**
Example: Interpret and present/communicate a message A basic pattern is interpreted and a corresponding message in symbolic form is presented. A basic pattern is decoded to a simple word, image or 3-word sentence/phrase. Emoji Secret Code Generator Worksheet - Primary Resource (twinkl.co.za) 	Learners can write their name with the emojis or a short instruction to their friends. Done in relation to C.6 and C.7
Assessment – Term 3	
Continuous Assessment – Refer to Section 4	

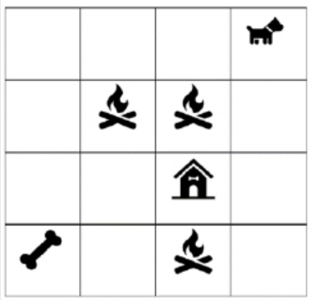
3.2.4 Term 4

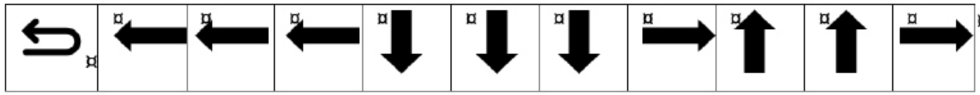
Content (Grade 1 / Term 4)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Reinforce from previous terms.** Use numbers, colours, shapes, body percussion, etc
C.7 Create or complete a pattern to represent a data set.	**Could be done with C.6**
Example:  DBE Grade 1 Workbook 2 (Maths) – p66	Learners create their own pattern, or they copy the pattern and then complete the pattern.
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Link to C.2, C.3, C.4 and C.5**
Example: Study the given map (grid) and accompanying code set on the right. • What word would SSB spell if he executes the code set? (Answer → BOAT) • Write down the code set that are required to have SSB spell SALT. • Which other words can SSB spell? 	This activity could be integrated with Language.
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.	
Example Learners code their names and an additional word such as 'hallo' to communicate a message using a coding card and give to their peers to decipher using the card. 	This can be integrated with phonics to e.g. build a three-letter word to illustrate the use of secret code.

Content (Grade 1 / Term 4)	Notes/Examples
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.1, C.3, C.4, C.5 C.6 and C.7**

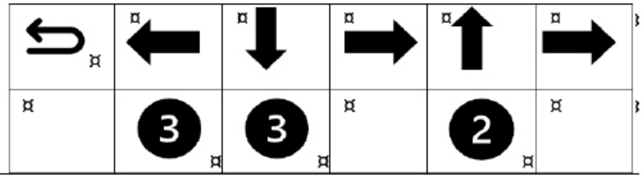
Example 1:
The dog needs to pick up the bone and return to the kennel. The code provided represents a sequential solution.
Rewrite the code provided to include repetition command where appropriate.

Code provided: ⮌ means turn around (dog first need to turn around to face opposite direction and start walking in the direction it is facing)





New, improved solution



Notes/Examples: Learners must now interpret and execute the code. The "robot" must follow instructions.
Learners reinterpret the code to change the solution from a sequential solution to one that uses repetition commands.

Note:
Where the *Turn Around* instruction is used, turning to face the direction into which the sprite/character must start walking, is not implied. Once the dog has turned around, the, turning left and right and facing the direction into which it must continue, are implied and the path as well as the orientation of the front of the character/sprite must be considered (the character/sprite faces the direction in which it will move next).

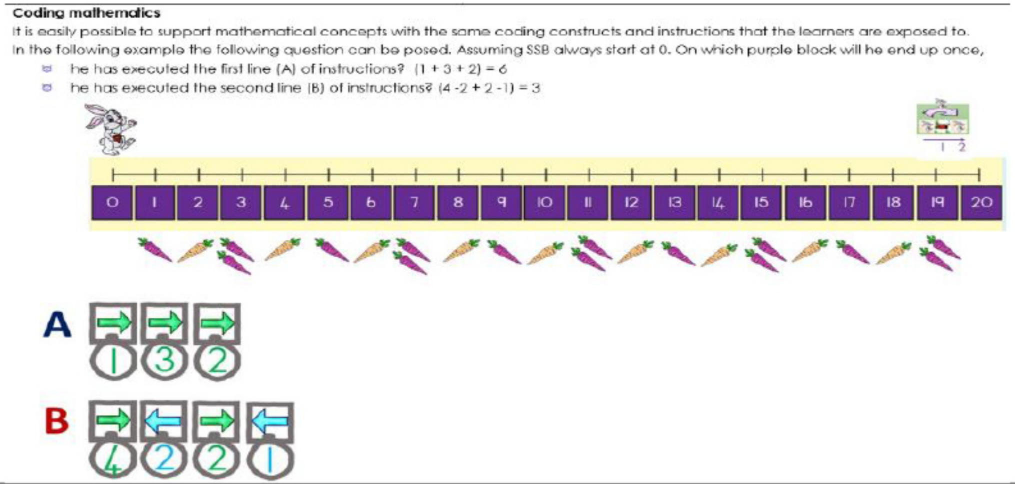
C.4 Debug a given symbolic or written set of instructions.	**Link to C.1, C.2 and C.3**
Provide learners with a grid and incorrect code. Learners need to determine where the problem is and correct the code	While executing, learners must make sure the instructions work. Debug if necessary and execute the code again until it is correct.
C.5 Evaluate a given solution towards potential improvement.	**Link to C.1, C.2, C.3 and C.4**

Example 1

Provide learners with a grid with more than one route to an object.

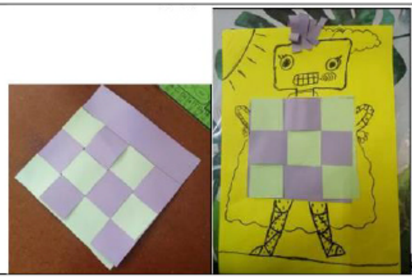
- Leaners need to identify all the routes.
- Leaners need to find the shortest route.

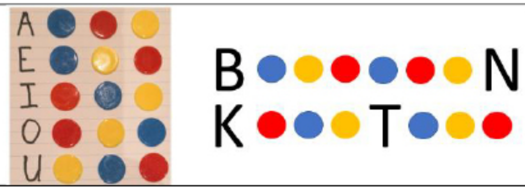
Example 2:



Notes/Examples: Give learners more than one solution that they must evaluate to find the most effective route/the route with the least steps.

Robotics	Could be integrated with Creative Arts
R.1 Explain what a robot is in simple terms.	**R.1, R.2, R.3 and R.4 can be done together**
R.2 Identify different types of robots.	**Reinforce from previous terms using different examples and activities**
Show learners pictures of different type of robots, e.g.	

Content (Grade 1 / Term 4)	Notes/Examples
• Robot vacuum cleaner (Type: household) • Robot assembling cars (Type: industrial)	
R.3 Outline the different components of a robot	**Link to R.1 and R.2**
Robots can have a body, arms and hands, sensors, a power source, wheels or legs and attachments Provide examples of different types of robots and identify their components	Reinforce from previous terms using different examples and activities
R.4 Present an understanding of how robots affect the world.	**Link to R.1, R.2 and R.4**
Example Ask learners how they think robots can help or harm us. Discuss possibilities such as robots doing work in dangerous places where humans cannot go, robots used in industry for assembling cars, etc.	Robots are changing the world by helping humans do things with greater efficiency and doing things that were not possible before
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.4**
Example The learners could apply paper weaving skills to create a mat. This mat can then be used in the design and creation of another artefact. 	Reinforce and extend from previous terms using different examples and activities.
R.6 Mimic the operations of a robot.	**Link to C.4**
Example: Coding cup challenge The person represents a robot that must move its arms to stack the cups http://www.youtube.com/watch?v=7zHVi16uXnU 	Reinforce and extend from previous terms using different examples and activities.
R.7 Create test and execute a set of robotic instructions.	**Link to C.4 and R.6**
Referring to R.6, test the operations for correctness	
Digital Concepts	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Reinforce and extend using different examples and activities.**
Example: Provide examples of technology, e.g. computers, smartphones, TVs, video games, 'robots' that perform specific tasks and discuss e.g. the following: Technology is designed with a purpose of solving problems that meet human needs and wants. It refers to tools, machines, or devices that make our lives easier or better. Technology is all around us, and we use it every day to communicate, learn, and have fun. The **purpose** of IT is to help people access and use information more easily and efficiently.	Information Technology (IT) is a type of technology that deals with information, such as data, images, and sound. IT includes things like computers, software, and the internet.

Content (Grade 1 / Term 4)	Notes/Examples
D.2 Recognise that he or she is living as citizens in a digital world.	**Reinforce and extend from previous Grades and terms**
Example: Possible questions for discussion: **Limit screen time:** • What is meant by limiting screen time? • Why it is important to limit screen time. • Why we also need family time, play time, time for schoolwork, exercise, etc **Interact appropriately with others online**: • What it means to be respectful and disrespectful online • When unkindness and disrespect turn into online bullying • How to stand up for others that are treated badly.	Reinforce from previous grades and terms using different examples and activities. Learners need to understand: • The concept of screen time, what it means and how do they balance their time with technology and other activities. • How to Interact appropriately with others when online
D.3 Demonstrate an understanding of the concept of a computing device.	**Reinforce and extend from previous Grades and terms using different examples and activities.**
D.4 Identify the common uses of ICT in the real world.	
D.7 Present a basic understanding of the concept of input processing and output.	
D.8 Interpret a pattern to represent or communicate a message or image.	**Link to C.6 and C.7**
Example: If the vowels are represented with the following 3-dot colour codes, what are the words on the right? 	
D.9 Create a pattern to represent or communicate a message or image.	**Link to C.6 and C.7**
Example If the vowels are represented with the following 3 dot colour codes how should each of the words be "coded". 	
Assessment – Term 4	
Continuous Assessment – Refer to Section 4	

Note:
In terms of coding, typically, problems could require learners to
- read code and explain what it does
- work through (trace) / act out code (physically or simulated) to determine the output or the correctness
- provide missing code instructions (code instructions are provided with some instructions or code elements missing) that learners need to complete
- translate verbal/written instructions (algorithm) to code.
- add some functionality/instructions to an existing program.
- rewrite a set of coding instructions to be more efficient, e.g. using a loop construct for code that is repeated
- choose the correct solution from 2-3 options
- compare different solutions to evaluate efficiency
- debug an algorithm or program (find the bug, describe the bug and correct it)
- develop a solution/algorithm (code instructions) based on a given problem or for an open-ended problem through planning, implementing, testing and debugging.

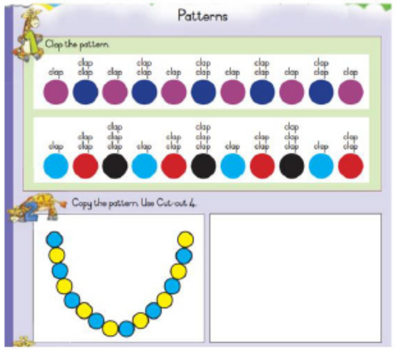
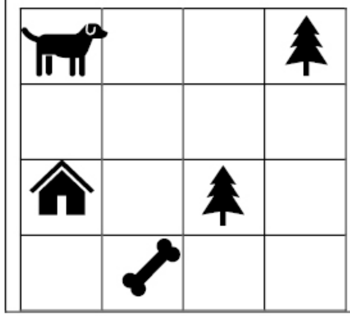
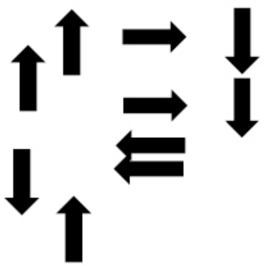
depending on the competency/(ies) the learner needs to demonstrate.

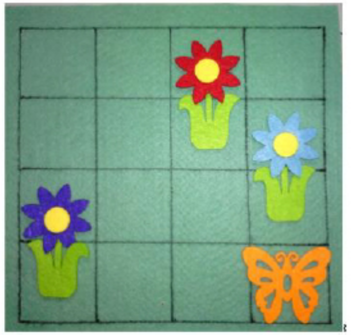
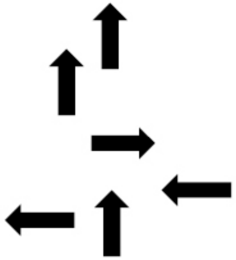
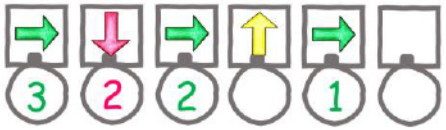
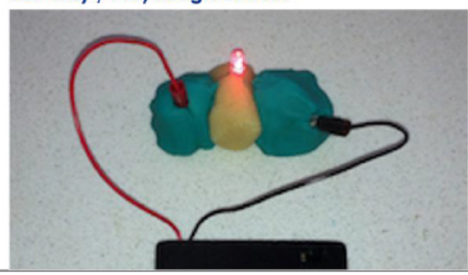
3.3 Grade 2

Note:

Teachers must include the following competencies and content in their Annual Teaching Plan (ATP), distributed across the terms and sequenced, organised and grouped (and integrated with other subjects in Grade R where applicable) in a manner that will facilitate learning ensuring ample retrieval and deliberate practise with feedback to ground principles and concepts, maximize the learners' learning outcomes and achievement but also ensure a gradual learning curve, and in a way that will make optimal use of time and resources.

3.3.1 Term 1

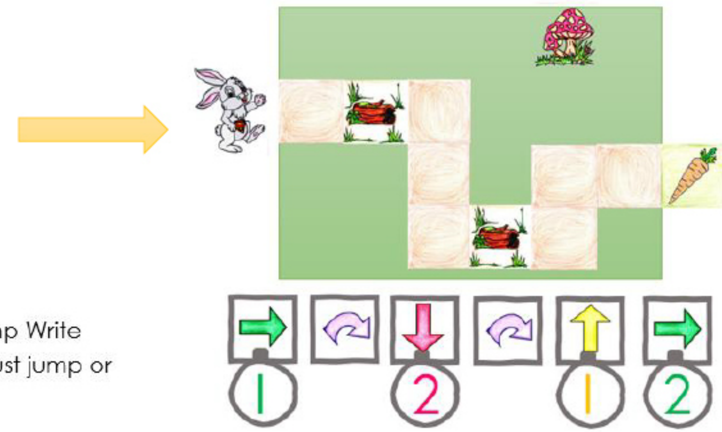
Content (Grade 2 / Term 1)	Notes/Examples
Pattern Recognition (± 2 hours)	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations	**Link to C.1**
 DBE Grade 2 Workbook 1 (Maths) – p56 Patterns can consist of numbers, colours, shapes, objects, movements, etc.	C.6 develops **pattern recognition** (part of computational thinking) that is eventually used to develop coding solutions as part of computational thinking to identify patterns in the coding problem and/or data by identifying similarities or differences that can help to solve the problem or refine the algorithm
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Link to C.2 and C.3**
Example: Develop instructions for the dog to first pick up its bone before going home. It must avoid the trees.  	One could extend the problem by asking learners how many different routes the dog can follow, then discuss which routes are the shortest (link to C.5). **Note** Evidence suggests that pupils should be taught – initially at least – in small bite-sized chunks. These steps in the learning process should be well-thought out and gradual as well as allow plenty of opportunity for practice (see, for example, Rosenshine, 2012; Coe *et al.*, 2014; Sealy, 2019).

Content (Grade 2 / Term 1)	Notes/Examples
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.	**Link to C.1**
Example: The orange butterfly wants to visit the red flower. How should it fly if it is not allowed to touch the blocks with the stems of the flowers.  	Learners present and explain their solution.
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.2 and C.1 and C.7**
Example: Interpret code and provide missing coding instructions Sipho Suber Bunny (SSB) need to collect the carrot at the end of the path. Study the grid on the right and the code provided below:  Provide the missing code instructions that will help SSB to collect the carrot Similar activities could also form part of code evaluation (C.7)) 	**Note** To ensure an acceptable learning curve/ progression, teachers can initially provide code that learners can choose from, e.g. provide learners with the following to code instructions to choose from to fill in the missing code, e.g.  As learners progress, they could be expected to fill in missing instructions without giving them instructions to choose from.
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	**Reinforce and extend from previous Grades using different examples and activities**
R.2 Identify different types of robots.	
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.6**
Example Learners could design a salt clay/playdough circuit. – they need to first complete a design process as represented in Grade 1 Term 3 and in R.7 below Electricity for kids - Easy Play Dough Circuits - Science Experiments for Kids (science-sparks.com) Electric Play Dough Project 1: Make Your Play Dough Light Up & Buzz! | Science Project (sciencebuddies.org) **Salt clay / Playdough Circuits** 	Reinforce and extend from previous Grades using different examples and activities. Learners first need to design their artefact. Refer to other R.5 examples that illustrates the design process.

Content (Grade 2 / Term 1)	Notes/Examples
R.6 Mimic the operations of a robot.	**Link to R.1, R.2 and R.5**
Example • Place the cones or other objects in the open space to create an "obstacle course" for the robot to navigate. • Divide learners into pairs and designate one member to be the 'robot' and the other to be the 'programmer. • Give the programmer a set of instructions for the robot to follow, such as "move forward three steps, turn left, move forward two steps, turn right, move forward four steps, turn around, etc." • The programmer should read the instructions aloud to the robot, who will then follow the instructions to navigate the obstacle course. • Once the robot completes the course, switch roles so that each learner has a chance to be the robot and the programmer. **Note**: As an extension activity, one can add more complex instructions or obstacles to the course, or have the learners create their own instructions for the robot to follow. One can also blind fold the 'robot' to see how well the programmer's instructions are designed.	This is an example of an outside "pair programming" activity. Explain to the learners that they will be pretending to be robots and following a set of instructions to complete a task, just like a real robot. This activity helps learners to develop the sequencing problem solving skills and encourages cooperative learning.
Digital Concepts	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Reinforce and extend from Grade 1.**
Possible discussion: **Technology** Technology refers to the tools, techniques, and processes that are used to create, develop, and improve products, services, and processes. Technology is used to solve problems, improve efficiency, and enhance productivity. **Purpose of IT:** Information technology (IT) is a specific branch of technology that deals with the storage, processing, and transmission of digital data. The purpose of information technology is to provide tools and resources that enable people to manage and use information effectively	Provide pictures of examples of technology and information technology. Discuss the concepts of technology and IT as well as their purpose
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1**
Possible discussion **Be respectful:** Always use kind words and treat others the way you would like to be treated. Remember that there are real people behind the screen, and what you say and do can have an impact on how others feel. **Be safe:** Never share your personal information, like your full name, address, phone number, or password, with anyone online. Also, never meet someone you've only talked to online without an adult present. Be safe: Never share your personal information, like your full name, address, phone number, or password, with anyone online. Also, never meet someone you've only talked to online without an adult present	Reinforce and extend from previous Grades using different examples and activities. Extend to: Interact appropriately with others when online. Learners need to note that it is important to remember to be kind and respectful to others, just like we would be in person. Empower learners to develop thinking skills and strategies to support the development of self-awareness and self-regulation so they learn to manage themselves as digital citizens
Assessment – Term 1	
Continuous Assessment – Refer to Section 4	

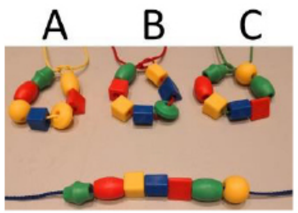
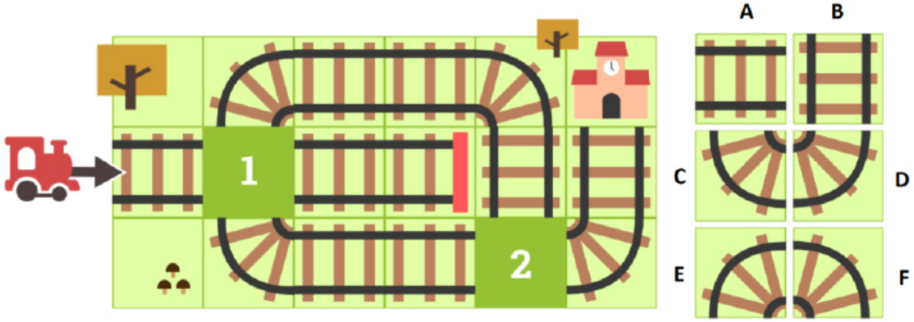
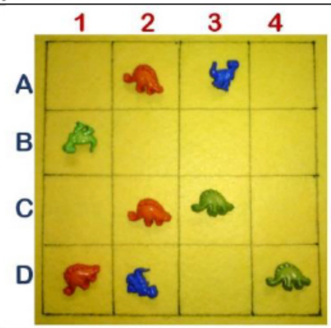
3.3.2 Term 2

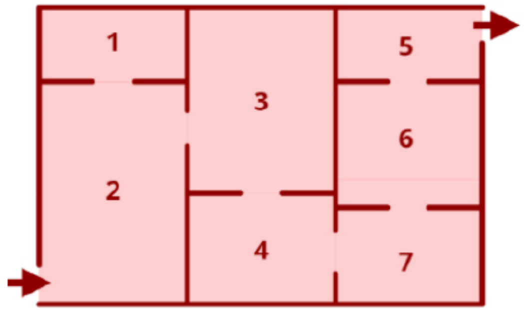
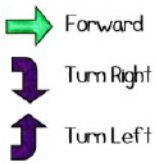
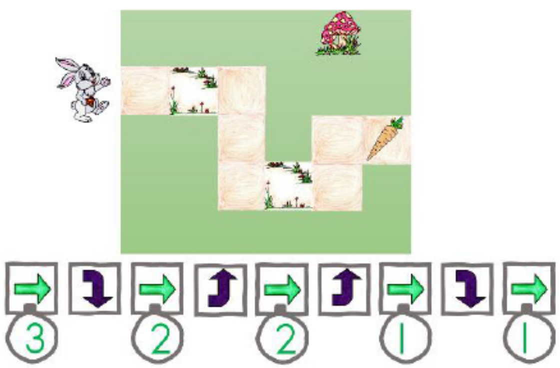
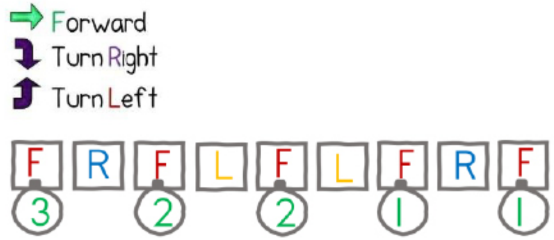
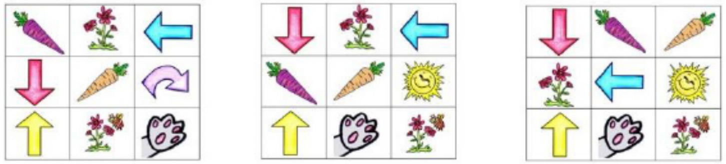
Content (Grade 2 / Term 2)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations	**Reinforce from previous Grades and Terms**
Example: There are four robot animals in a shop: a bear, a bird, a rabbit and a cat. One robot animal secretly walked around the shop at night. A trail of footprints was left on the floor. Which robot animal left the footprints? 2021-TS-Elementary-Question-Paper.pdf (olympiad.org.za) 	Gradually increase the complexity of patterns.
Algorithm Design and Coding	
C.1 Apply computational thinking skills to develop a set of logical instructions to solve a problem.	**Link to C.2 and C.3**
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.	This activity uses C.1 and C.2
Example The butterfly must reach the flower using an instruction set with a restriction of 6 instructions and a grid of max 4x4 A turn left arrow  is introduced (and by implication also the turn right)  The Turn right could be introduced moving the butterfly one block left. 	**Note** For Grade 2 and Grade 3, whenever the character/sprite is instructed to turn left or right, the orientation of the front of the character/sprite should be considered. This implies that the character/sprite face the direction in which it will move next).

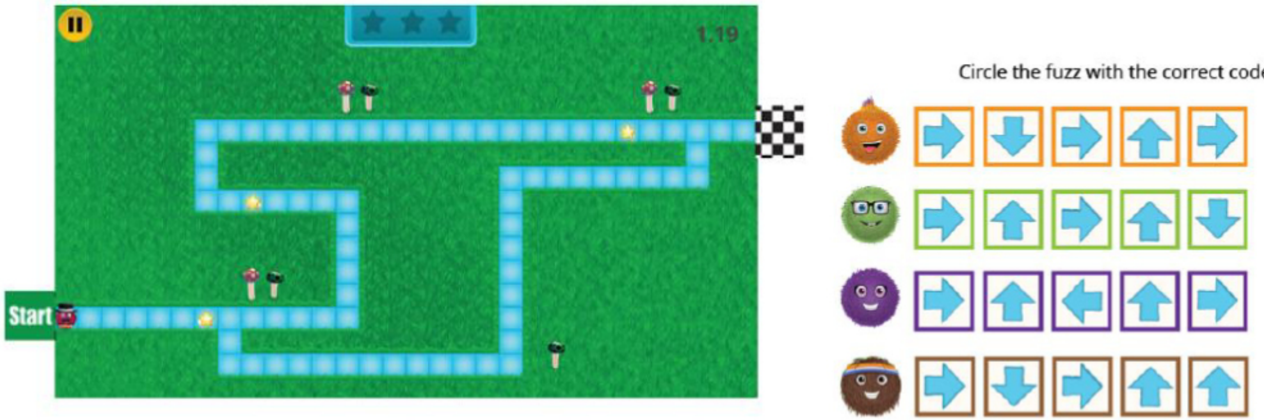
Content (Grade 2 / Term 2)	Notes/Examples
C.3 Interpret and execute a given symbolic or written set of commands	
SSB can jump over logs. When he jumps, he moves 2 spaces (tiles) forward.  Sipho can only move one block at a time and jump Write down the instructions and how many times SSB must jump or move to get to the carrot. Sipho can only move one block at a time and jump Write down the instructions and how many times SSB must jump or move to get to the carrot.  	Present the solution with directional coding cards and using the repetition structure.
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	**Reinforce and extend from previous Grades and Term using different activities and examples.**
R.2 Identify different types of robots.	
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.1 and R.2**
Example Using a plastic spoon, rubber bands and ice cream sticks, build a spoon catapult. See link for design instructions, https://stlmotherhood.com/popsicle-spoons-catapult-challenge/ 	Reinforce from previous grades and terms using different activities.
R.6 Mimic the operations of a robot.	**Link to C.1, C.2, C.3 and R.1, R.2 and R.5**
Example: Divide the learners into groups and provide them with a sheet of paper to write down their instructions. Have the learners discuss and plan out a simple sequence of instructions that a robot could follow, such as moving forward, turning left, and picking up an object. Once the learners have planned out their sequence of instructions, have them write down their instructions step by step on the paper. After writing out their instructions, have the learners test their instructions by having a group member act out the instructions. This will help the learners identify any errors or missing steps in their instructions. Finally, have the learners revise and refine their instructions until they can successfully complete their robot task	Learners develop their own set of instructions to mimic the operations of a robot.

Content (Grade 2 / Term 2)	Notes/Examples
Digital Concepts	
D.1 Outline the concept of technology and purpose of information technology (IT).	**Reinforce and extend from previous Grades and Terms using different examples and activities.**
Possible discussion: What does technology refer to? What is technology used for? What is the purpose of IT? What is IT used for?	
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1**
Possible discussion **Be honest**: Always tell the truth when you're online. It is important to be honest about who you are and what you're doing. **Be a good friend**: Just like in person, it is important to be a good friend to others when you are online. This means being there for them, helping them when they need it, and standing up for them when they are being treated unfairly.	Reinforce and extend from previous terms. Interact appropriately with others when online.
D.3 Demonstrate an understanding of the concept of a computing device.	**Link to D.2**
Possible Activity Explain what a computing device is and what it does. Explain that a computing device is a machine that can process and store information and can be used to do things like play games, watch videos, and communicate with others. Show the learner pictures of different computing devices and ask them to identify what they are. Talk about the differences between the devices, such as size, shape, and features. Give the learner a piece of paper and ask them to draw a picture of a computing device. Encourage them to be creative and include details like buttons, screens, and keyboards. Once the drawing is complete, ask the learner to explain what their computing device does and how it works. Have them describe what they might use the device for, such as playing games, doing homework, or talking to friends and family How to take care of computing devices and use them safely.	Reinforce and extend from previous Grades and Terms using different examples and activities.
Assessment – Term 2	
Continuous Assessment – Refer to Section 4	

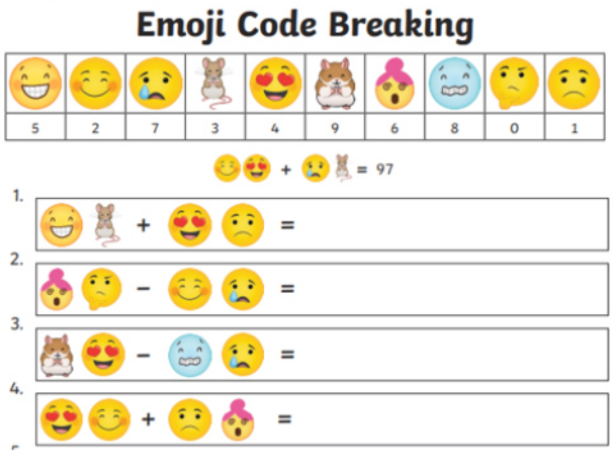
3.3.3 Term 3

Content (Grade 2 / Term 3)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Link to C.7**
 Which one of the bracelets (A, B or C) is in the same order and pattern as presented in straight line pattern below?	Gradually increase the complexity of patterns.
C.7 Create or complete a pattern to represent a data set.	**Link to C.6**
Example 1: The train must get to the station; however, some track pieces are missing. Study the picture and indicate which two track pieces from the pieces on the left (A, B, D, E or F) should be placed in position 1 and position 2.  2021-TS-Elementary-Question-Paper.pdf (olympiad.org.za)	Learners can also create their own patterns, based on the problem, or the teacher can give them an incomplete pattern with some of the elements missing. They must then complete the pattern by adding the missing elements.
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Reinforce and extend from previous Grades and terms**
Example: Study the following placement and answer the questions that follows. • Which row has only one dinosaur? • What colour of dinosaur occur twice in the same column? • Write down the coordinates of all the green dinosaurs 	Introduce learners to the concept of coordinates on a grid. For C.1 to C.3 problem-based questions and scenarios can be introduced, where scenarios are given, and the learners are expected to answer questions based on the scenario.

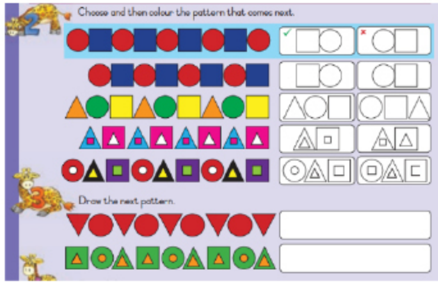
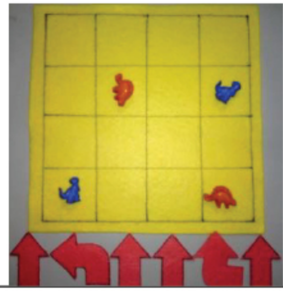
Content (Grade 2 / Term 3)	Notes/Examples
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition and conditional constructs.	**Link to C.1**
Example: Museum visit Visitors visiting the museum are only allowed to go through all the rooms exactly once. This is called a one-way tour. Therefore, the following restrictions apply: **Restrictions** They may not visit a room more than once. They are also not allowed to use the same door for entering and exiting a room. The visitors must start at the arrow that enters the museum and leave by way of the door with the arrow leaving the museum. **Task** Use directional coding cards and pack the instructions according to the rules. Adapted from 2021-TS-Elementary-Question-Paper.pdf (olympiad.org.za) 	Complexity is increased by adding rules or restrictions
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.1, C.2, C.6 and C.7**
Example SSB can only perform the following commands.  Sipho Super Bunny can only move one block at a time. Write down the instructions and how many times SSB must move and turn right or left to get to the carrot. 	They now execute the code to test it. Learners should also be able to replace the symbols with characters to represent the actions. 
C.4 Debug a given symbolic or written set of instructions.	**Link to C.1, C.2 and C.3**
Example 1 Study the following 3 grids with symbols and answer the questions on the right.  A B C Explain which grid(s) break the rule for the limitation(s) provided below: 1. There may not be two arrow symbols in the same row. (Answer C) 2. Flowers may not appear in the same column (Answer is A). 3. There may not be any bow arrow in the grid (Answer A)	Learners follow different instruction sets, of which some are incorrect, to find the correct solution, then they need to explain why the error in the incorrect ones.

Content (Grade 2 / Term 3)	Notes/Examples
Example 2: Study the problem below, then find the correct set of instructions on the right. 	
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	R.1, R.2, R.3, R.4 and R.5 can be done together First learners explain what a robot is and name different types of robots as well as the different components of a robot and how robots affect the world. The above can give them inspiration for designing their own robot. Then they can ideate about the type of their robot, what components it will have, what it can do and how it will impact the world
R.2 Identify different types of robots.	
R.3 Outline the different components of a robot	
Body - usually made up of metal or plastic and is the outer covering that holds all the other components together. **Arms and hands** – -many robots have arms and hands that can move and manipulate objects. These arms and hands are usually made up of joints that allow them to bend and move in different directions. **Sensors** - robots have sensors that allow them to "see" and "hear" the world around them. These sensors can include cameras, microphones, and other devices that can detect things like light, sound, and movement. control system - is what allows the robot to "think" and make decisions. It's usually made up of a computer or microcontroller that's programmed with instructions for how the robot should behave. **Power source** - robots need a source of power to operate. This could be a battery or a plug that connects the robot to an electrical outlet. **Wheels or legs** – some robots have wheels or legs that allow them to move around. These wheels or legs are usually controlled by the control system. **Tools or attachments** – depending on the task the robot is designed to do, it may have various tools or attachments, such as a vacuum cleaner attachment, a paintbrush, or a gripper for picking up objects.	
R.4 Present an understanding of how robots affect the world.	
R.5 Design a simple product (artefact) based on a set of design specifications.	
Example: Linking to **Language**, learners can present their robots and tell the class about their robots: • What type of robot it is • What components it has • What it can do • How it can impact the world	

Content (Grade 2 / Term 3)	Notes/Examples
R.6 Mimic the operations of a robot.	**Link to C.3 and R.7**
Example: Learner pretends that he or she is a robot that has been programmed to pick up and sort objects by colour and follow the following steps: 1. Pretend to turn on the power (e.g. Battery) 2. Use their sensors to "see" the objects that need to be sorted. We might have a camera or other device to detect colours. 3. Using their arms and hands, pick up one object at a time and examine it to see what colour it is. If it's blue, we would place it in the "blue" pile. If it's red, we would place it in the "red" pile, and so on 4. Repeat this process until all the objects have been sorted into their respective piles. 5. Once finished sorting, sensors used to check and make sure that all the objects have been sorted correctly. If there are any errors, pick up the misplaced objects and move them to the correct pile. 6. Turn off the power to conserve energy for the next task	Learners following a set of "programmed" instructions to complete a task, using different parts and systems to accomplish it. Reinforce the decision structure (IF... THEN), e.g. As question: What is the colour of the object picked up? • IF object picked up is red, put with red pile. • IF object picked up is blue, put with blue pile.
R.7 Create, test and execute a set of robotic instructions.	**Link to R.6 and C.4**
Example Use the activity from R.6 above and let learners now create code using coding cards to create instructions for the robot to pick up objects.	Extend example in R.6
Digital Concepts	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT). **D.2 Recognise that he or she is living as citizens in a digital world.**	**Reinforce and extend from previous Grades and Terms using different examples and activities.**
Being a digital citizen means being a good and responsible person when we use the internet and technology. When we're digital citizens, we follow rules and guidelines to make the online world a safe and a positive place for everyone such as to treat others with kindness and respect, think before we post.	Reinforce and extend from previous Grades and Terms using different examples and activities. Interact appropriately with others when online
D.3 Demonstrate an understanding of the concept of a computing device. **D.4 Identify the common uses of ICT in the real world.**	**Reinforce and extend from previous Grades and Terms using different examples and activities.**
D.5 Differentiate between the components of an ICT system.	**Link to D.3 and D.4d**
Possible discussion An ICT system is a system that uses computers, programs (instructions), and networks (communication between computing devices) to process and store information. It has distinct parts that work together to help us use computers to do different tasks: • The first part is the computer itself. • The second part is the instructions (software) that tell the computer what to do. • The third part is the data. Data is the information that we put into the computer to work with. This can be things like pictures, music, or text. • An ICT system also has networks. Networks are like the roads that help computers talk to each other. They connect computers together so that we can communicate, share information and work on projects together.	Use pictures to discuss the different components of an ICT system. Use examples that are age appropriate and that learners will understand, e,g, a mobile phone system
D.7 Present a basic understanding of the concept of input processing and output.	**Link to D.3 and D.5 and D.8**
Example – IPO Divide learners into groups of 3. One learner acts as 'input', another learner acts as 'processing' and the third learner actst as 'output' Provide different colours of play dough and a cookie cutter for each group. 'Input' collects the dough (input) and passes it to 'Processor'. 'Processor' mixes the dough and use the cookie cutter to cut 'cookies' from the mixed dough. Then 'output' collects the 'cookies' from 'processor' and show them to the class as the processed product.	
D.8 Interpret a pattern to represent or communicate a message or image.	**Link to C.6 and C.7 and D.7**

Content (Grade 2 / Term 3)	Notes/Examples
Example – Coding and Decoding  https://content.twinkl.co.uk/resource/35/a6/t2-m-17313-ks2-emoji-code-breaking-activity-sheets-english_ver_1.pd	An activity such as the example can also be used to illustrate Input, Processing and Output: • **Input**: Prepare a simple cipher code (secret code) where symbols/numbers replace letters of the alphabet. Write a short message in the code as the "input" for the computer. • **Processing**: Provide the learner with a decoder chart to figure out the symbols and translate the message and do the calculation. This represents the computer processing the encrypted/coded data. • **Output** Once the message is decoded, the child can read it out loud as the computer's "output."
Assessment – Term 3	
Continuous Assessment – Refer to Section 4	

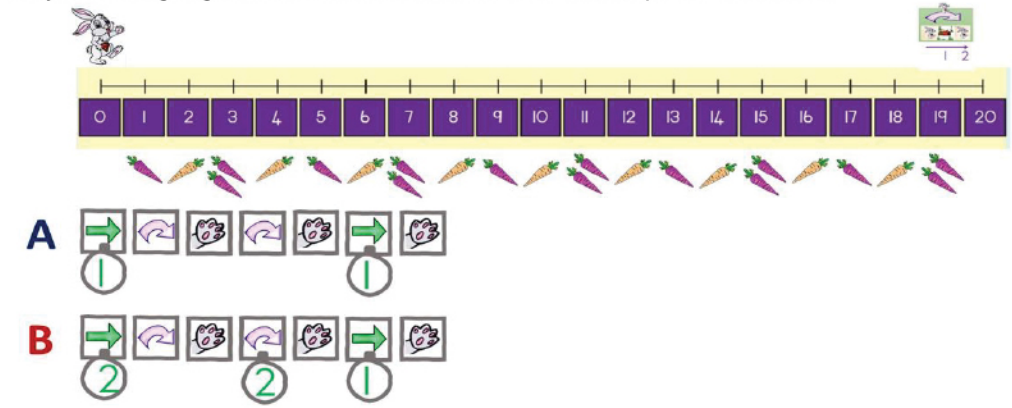
3.3.4 Term 4

Content (Grade 2/ Term 4)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations. **C.7 Create or complete a pattern to represent a data set.**	Reinforce from previous Grades and terms using different types of patters, e.g. numbers, colours and/or shapes, objects, poems, physical activities, etc.
Examples:  DBE Grade 2 Workbook 1 (Maths) – p58	
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Link to C.2 and C.3**
Example: In this example code should be written which includes directional indicators for each step and single movement instructions where the arrow indicates the movement in a single direction (forward 1). The solution is for the orange dinosaur to meet up with his fellow same colour dinosaur. Note: There may be some alternates as well. These alternate solutions should also be accepted and discussed. The complexity can be increased by instructing the learners to write two sets of code for each colour dinosaur to visit his/her friend, but the paths may not cross. 	Gradually increase complexity by extending the grid, e.g. adding obstacles and limitations Evidence suggests that pupils should be taught – initially at least – in small bite-sized chunks. These steps in the learning process should be well-thought out and gradual as well as allow plenty of opportunity for practice (see, for example, Rosenshine, 2012; Coe *et al.*, 2014; Sealy, 2019).
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.	**Link to C.1**
Examples Introduce the concept of a conditional coding construct (IF…THEN) using everyday decisions, at the level of the learner. DBE Grade 2 Workbook 1 (Life skills) – p41  	The concept of a conditional/decision construct should be introduced based on daily decisions that is familiar to learners before introducing it as part of a coding solution. **Note:** Ensure a gradual learning curve. When introduced, start with one decision structure only (also do not initially combine with a repetition structure – only sequence and one decision

Content (Grade 2/ Term 4)	Notes/Examples
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.1 and C.2**

Example 1:

Study the following diagram and instructions that follows and answer the questions that follows.



Note:
The instruction, pickup, picks up all the carrots at that position on the number line. A Jump skips one position to the right.

Note:
The example introduces the learners to the concept of tracing code in an elementary way.

a) Examine the code for instruction set A:

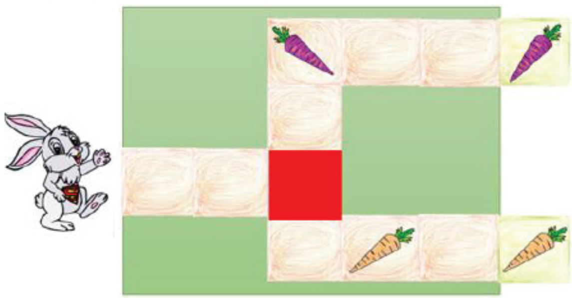
- After the code has executed on which purple number on the number line will SSB end up? (Answer 6)
- Atter the code has executed how many purple carrots will SSB have collected? (Answer 3)
- Atter the code has executed how many orange carrots will SSB have collected? (Answer 1)

b) Examine the code for instruction set B:

- After the code has executed on which purple number on the number line will SSB end up?
- Atter the code has executed how many purple carrots will SSB have collected?
- Atter the code has executed how many orange carrots will SSB have collected?

Example 2: Making decisions

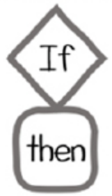
Study the picture below:



Note:
Example 2 combines sequence, decision and repetition structures.

Note

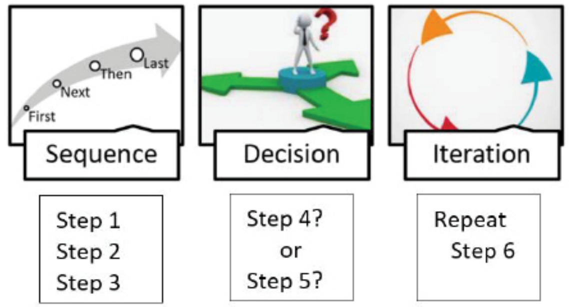
The following symbol is used to represent an IF…THEN coding construct.



Introduce the decision construct (IF…THEN…)

Coding requires learners to master the basic coding constructs:

- Sequence
- Repetition/Iteration (loops)
- Decision



A **sequence** is a set of instructions or commands that are executed one after the other, in order. It's like following a recipe step by step.

Decision construct is like making choices. It allows the computer to execute a different instruction or set of instructions based on the outcome of the decision: whether the condition result in true of false,

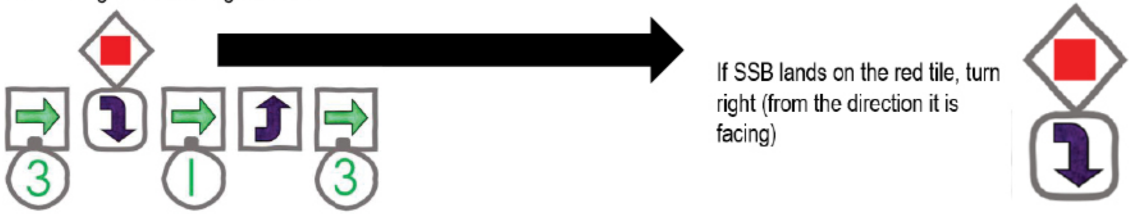
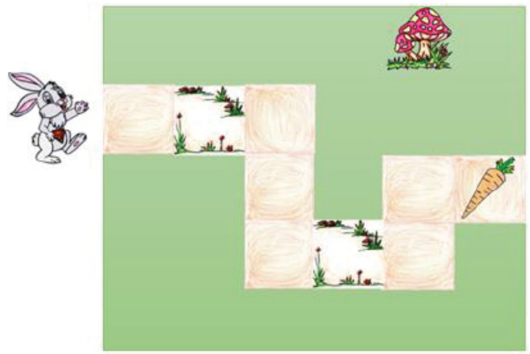
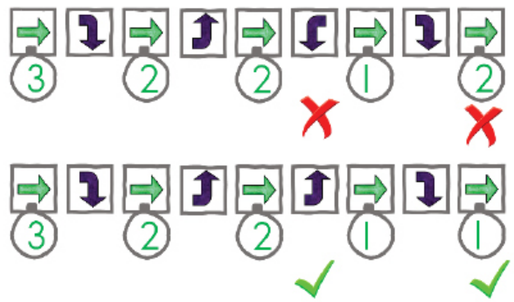
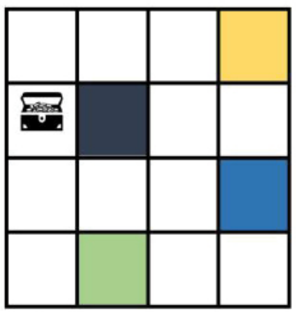
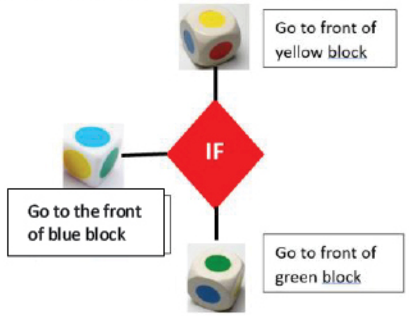
Repetition, or loops, allows the computer to repeat a set of instructions multiple times. It's like doing something over and over without having to write the same instruction(s) multiple times

Note:

Keep to an acceptable learning curve – gradually increase complexity, e.g.

- One decision as part of a basic problem
- Then two decisions as part of a basic problem
- Then only begin to combine a decision structure with a repetition structure.

Literature suggests that the biggest problem of novice programmers does not seem to be the understanding of basic coding concepts [n isolation] but rather learning to apply them [and combine them]. Therefore, at this level, beware of giving learners programming tasks that combine too many concepts (Robins, 2019).

Content (Grade 2/ Term 4)	Notes/Examples
SSB does not like purple carrots. When it lands on the red tile, it must decide which direction to go to avoid the purple carrots and to collect only the orange carrots. The algorithm below represents the instructions SSB must follow (including the direction it must turn when it lands on the red tile) to avoid the purple carrots and get to the orange carrots  If SSB lands on the red tile, turn right (from the direction it is facing)	Typically, problems could require learners to provide missing code instructions (code instructions are provided with some instructions or code elements or require learners to develop the solution algorithm for themselves.
C.4 Debug a given symbolic or written set of instructions.	**Debugging relates C.1, C.2 and C.3**
Example: There are 2 errors in the solution provided. Identify and correct the errors.  	It is important for learners to be able to debug a solution. Debugging code is part of the coding process.
C.5 Evaluate a given solution towards potential improvement.	**Link to C.1, C.2, C.3 and C.4**
Example: Three learners play a game using a grid and a die with coloured dots and a pile of directional coding cards. The objective of the game is to see which learner can get to the treasure box first.   **Materials required**: Grid, die, treasure, Pile of directional coding cards. Directional cards include several of the following: Forward, Turn Right, Turn Left, Turn Around (facing opposite direction, Backwards) How to play • Learners throw a die with colours on each side. • If colour throwed, is yellow, learner starts in front of yellow block on the grid.	Leaners develop their own solutions based on the cards they draw. Learners need to continually present their coding solutions by interpreting the coding cards and evaluate the cards in terms of the route they can build with the coding instructions they have and weather the cards they have will lead them to the treasure.

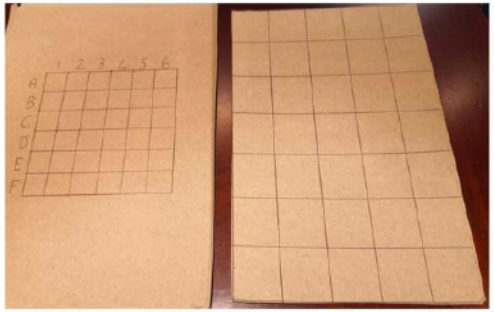
Content (Grade 2/ Term 4)	Notes/Examples
• If colour throwed, is blue, learner starts in front of blue block on the grid. • If colour throwed, is green, learner starts in front of green block on the grid. • Now each play gets a turn to throw the die again. • If the die indicates the learner's colour (the block where the need to start), they draw one directional card from the pile • Each time they collect a directional card from the pile, they check to see if they can reach the treasure, else they need to wait for their next throw. • If a learner lands on the black block the learner must go back to the coloured block where they started. The first learner that gathers the right cards to get to the treasure, wins.	
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	**R.1 and R.2 could be done together**
R.2 Identify different types of robots.	Learners need to understand that a robot is a machine built by humans and programmed to perform tasks, which a human can also do, e.g. a robot vacuum cleaner. A robot can thus substitute a person, performing a task that the person could do. Robots can only do what they are programmed to do (follow a set of instructions).
Provide examples of robots and non-robots and discuss these. Provide examples of different types of robots and discuss these.	
R.3 Outline the different components of a robot	**Link to R.1 and R.2**
Body - usually made up of metal or plastic and is the outer covering that holds all the other components together. **Arms and hands** – -many robots have arms and hands that can move and manipulate objects. These arms and hands are usually made up of joints that allow them to bend and move in different directions. **Sensors** - robots have sensors that allow them to "see" and "hear" the world around them. These sensors can include cameras, microphones, and other devices that can detect things like light, sound, and movement. **Control system** - is what allows the robot to "think" and make decisions. It's usually made up of a computer or microcontroller that's programmed with instructions for how the robot should behave. **Power source** - robots need a source of power to operate. This could be a battery or a plug that connects the robot to an electrical outlet. **Wheels or legs** – some robots have wheels or legs that allow them to move around. These wheels or legs are usually controlled by the control system. **Tools or attachments** – depending on the task the robot is designed to do, it may have various tools or attachments, such as a vacuum cleaner attachment, a paintbrush, or a gripper for picking up objects.	Components such as: body, arms and hands, sensors, control system, power source, wheels or legs, tools or attachments.
R.4 Present an understanding of how robots affect the world.	**Link to R.1, R.2 and R.3**
Possible discussion: **Helping with Work:** Robots can help us with different kinds of work. For example, in factories, robots can assemble things like cars or toys very quickly and accurately. **Making Life Easier:** Some robots are designed to help us with everyday tasks. For instance, there are vacuum robots that can clean our floors by themselves. It is also important to remember that while robots can do many amazing things, they are created by humans and need to be programmed and controlled by us. They are designed to help us and make our lives better, but it is still humans who decide how they are used and make sure they are safe.	Reinforce and extend from previous Grades and terms using different examples and activities.
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.6**
Example: Design a board game	The learners design the board game in R.5, then mimic the operations of a robot (this board game) in R.6

Content (Grade 2/ Term 4)	Notes/Examples

Materials required:

3 wooden blocks to create 3 different dices (Dice 1 has the values A to F, Dice 2 has the values 1 to 6 and Dice 3 has the codes: ***F, F, TR, TL, Bon*** and ☹

a. The F represents Forward.
b. TR represents Turn right.
c. TL represents Turn left.
d. Bon represents a bonus throw and
e. ☹ represents skip a card/turn.



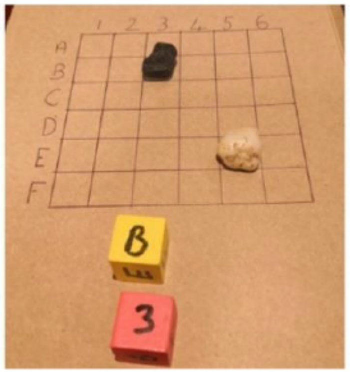
- A reused cardboard from a cereal box with a 6 x 6 grid and the indicators A to F and 1 to 6 for the rows and columns respectively
- A set of coding cards made from the remainder of the board with the codes TR, TL and F appearing twice the number of times. (The coding cards are cut out)
- 3 small stones to represent obstacles.
- A one Rand coin to represent a prize (it may be substituted with a sweet or any other indicator)
- Two player blocks representing an icon/charm for each player (in this example two uni-fix counting blocks, one red and blue are used)
- Small coding cards made from the remainder of the carton of the cereal box.

R.6 Mimic the operations of a robot.

Example: Referring to the board game designed in R.5

How the game works.

1. The three obstacles (i.e., stones in our example) are placed randomly on the grid by throwing the two dice with the letter and the digit together, e.g. the letter die indicates the row, say, D and the number die indicates the column, say for 4. The obstacle will then be placed on block D.4. This action is repeated until all three stones are placed on the board; *A stone may not be placed on the same position twice*.
2. Now, place the prize using the same process as in 1. Above



3. Now, the starting position for each player is determined by throwing the single row-and-column die.

Notes/Examples – Link to R.5 and R.7

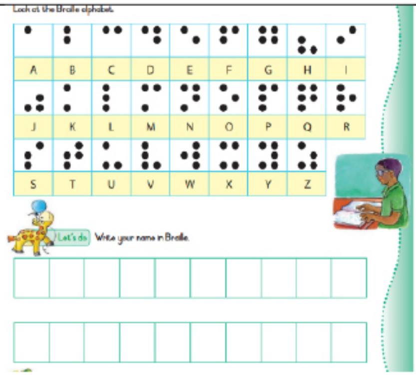
Using the board game designed in R.5, learners will now mimic how a robot responds to instructions.

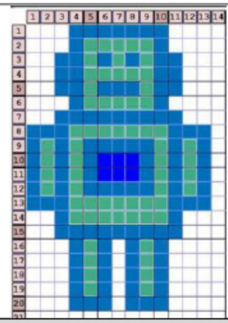
Coding constructs that are reinforced with this activity:
Repetition:
Throw letter die and number die to identify the block where the first obstacle needs to be placed.
REPEAT this process **UNTIL** all three obstacles are placed (**REPEAT 3 times**)

Note:
In this activity there are several instances where repetition is used. This must be pointed out to learners and emphasised.

There are also several instances where decisions are required. Point out to learners and emphasise.

Content (Grade 2/ Term 4)	Notes/Examples
Player RED throws D and is placed on D on the left of the grid  Player BLUE throws 6 and is placed on the number 6 on top of the grid  (A rule may be added that if the player is within six moves of the prize the player icon should be moved to the opposite outside edge. In the case for blue, it would have been at the bottom outside of column 6) (The rule was not applied in this case) **Moving the players** • Using the ***move*** dice, each player throws 6 times and collects the applicable corresponding coding card. If the ☹ is thrown, then a turn is missed. • Using the coding cards at their disposal, the players must then pack out the code to have their icon move to the prize. • If the player does not have the correct cards, he or she may throw the dice again to get an additional coding card. This is alternately repeated until one of the players packs out the correct solution to have their icon reach the prize. • Each opponent player must verify that the player indeed has a correct solution or not. • The winner is the one who reaches the prize first	
R.7 Create, test and execute a set of robotic instructions.	**Link to R.6**
Using the board game designed in R.5, and played in R.6: Once a learner has collected the six coding cards, the learner needs to pack out instructions (create a set of instructions) to move to the prize. In doing so, the learner needs to test/evaluate to see if the cards he/she has, will be able to move his/her charm to the prize, IF not, THEN the learner need to throw again and repeat the packing and testing/evaluation process UNTIL he/she	The last few steps of the game in R.6 are applicable to creating, testing and executing the instructions.
Digital Concepts	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	Reinforce and extend from previous Grades and Terms using different examples and activities.
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1 and D.2**
Interact appropriately with others when online. Discuss with learners why and how they must interact appropriately when online.	Reinforce and extend from previous Grades and Terms using different examples and activities. Empower learners with thinking skills and strategies to support the development of self-awareness and self-regulation so they learn to manage themselves as digital citizens
D.3 Demonstrate an understanding of the concept of a computing device.	**Link to D.1, D.2 D.4, D.6 and D.7**
Example: As an extension from D.3 activity in Term 2: The learner creates a "poster" with pictures of different computing devices and labels that describe their features and functions. This can be a great way to reinforce the concept of a computing device and help the learner understand the different types of devices that are available. Learners can also explain: • What the devices can be used for	Reinforce and extend from previous Terms using different examples and activities

Content (Grade 2/ Term 4)	Notes/Examples
• What impact they have in our lives (D..6) • How to take care of computing devices and use them safely • How to behave when using the devices (D.2) • How these devices receive input and how they give output	
D.4 Identify the common uses of ICT in the real world.	**Link to D.1, D.3 and D.5**
Examples: **Communication**: social media, e-mail, messaging apps **Education**: Digital learning material **Business**: Point-of-sales **Entertainment**: streaming videos and music	Identify and discuss common uses in areas of communication, education, business, and entertainment
D.5 Differentiate between the components of an ICT system.	**Reinforce and extend**
D.6 Explain how the adaptation of technology impacted the world we work and live in.	**Link to D.4 and D.5**
Reinforce from previous grades and terms. **Impact**: (different sectors), e.g. • **Communication**: Made the way we communicate much faster and easier (Instant messaging, e-mail, etc.) • **Education**: Changed the way we learn (online resources)	Reinforce and extend from previous Grades and Terms using different examples and activities.
D.7 Present a basic understanding of the concept of input, processing and output.	**Link to D.8 and Cs**
Example activity: Secret Code Decoder: Divide learners into small group of three learners and provide each group a simple cipher code chart where symbols replace letters of the alphabet. One learner acts as input, second learner acts as processing and third learner acts as output. • **Input**: First learner prepares input using the cipher code and writes e a short message in the code (encrypted message) and passes it to the second learner as the "input" for the computer to be processed. • **Processing**: Second learner uses the cipher code chart to figure out the symbols and translate the message. This represents the computer processing the encrypted data. The translated message is then passed it to the third learner • **Output**: Third learner reads the translated/decrypted message out loud as the computer's "output."	Reinforce and extend using different examples and activities Initially, the words *translate to* and *translate back* could be used to introduce learners to the concepts encoding and decoding in programming. Learners are then gradually exposed to the words *encode* and *decode*
D.8 Interpret a pattern to represent or communicate a message or image.	**Link to D.9, C.6 and C.7**
Example: Encoding Learners use the braille chart to write their names in braille. DBE Grade 2 Workbook 2 (Life skills) – p52 This example from the DBE workbook could be done where a learner is required to encode and decode a message. (This then links to D.9) 	An activity such the example can also be used to illustrate Input, Processing and Output (D.7): • **Input**: Prepare a simple cipher code (secret code) where symbols/numbers replace letters of the alphabet. Write a short message in the code as the "input" for the computer. • **Processing**: Provide the learner with a decoder chart to figure out the symbols and translate the message. This represents the computer processing the encrypted/coded data. • **Output** Once the message is decoded, the child can read it out loud as the computer's "output."

Content (Grade 2/ Term 4)	Notes/Examples
D.9 Create a pattern to represent or communicate a message or image.	**Link to D.8**
Example 1 – Use a grid with blocks to code 'message' Divide learners into pairs. Provide each pair with an empty grid as shown on the right (without the robot) and a set of instructions (containing coordinates (row, column) to specific blocks that must be coloured a specific colour. and a colour that the. The instructions navigate learners to code a robot or a computing device. One learner act as the navigator (reading the instructions) and the other learner as the driver (following the instructions) Once done, pairs show their 'pattern message; to other pairs to 'read' and explain what the message is communicating. 	This activity can be extended where learners code their own message such as a smiley to e.g. communicate how they felt when the woke up in the morning
Assessment – Term 4	
Continuous Assessment – Refer to Section 4	

Note:

In terms of coding, typically, problems could require learners to

- read code and explain what it does or
- work through (trace) / act out code (physically or simulated) to determine the output or the correctness or
- provide missing code instructions (code instructions are provided with some instructions or code elements missing) that learners need to complete or
- translate verbal/written instructions (algorithm) to code.
- add some functionality/instructions to an existing program.
- rewrite a set of coding instructions to be more efficient, e.g. using a loop construct for code that is repeated or
- choose the correct solution from 2-3 options or
- compare different solutions to evaluate efficiency or
- debug an algorithm or program (find the bug, describe the bug and correct it)
- develop a solution/algorithm (code instructions) based on a given problem or for an open-ended problem through planning, implementing, testing and debugging.

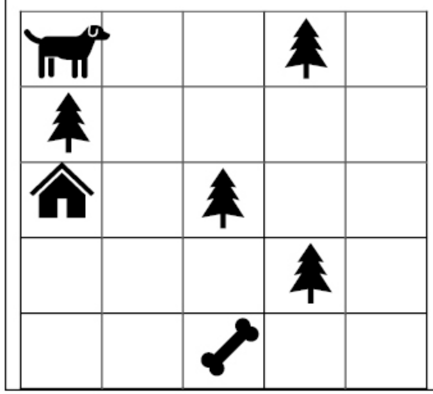
depending on the competency/(ies) the learner needs to demonstrate.

3.4 GRADE 3

Note:

Teachers must include the following competencies and content in their Annual Teaching Plan (ATP), distributed across the terms and sequenced, organised and grouped (and integrated with other subjects in Grade 3 where applicable) in a manner that will facilitate learning ensuring ample retrieval and deliberate practise with feedback to ground principles and concepts, maximize the learners' learning outcomes and achievement but also ensure a gradual learning curve, and in a way that will make optimal use of time and resources.

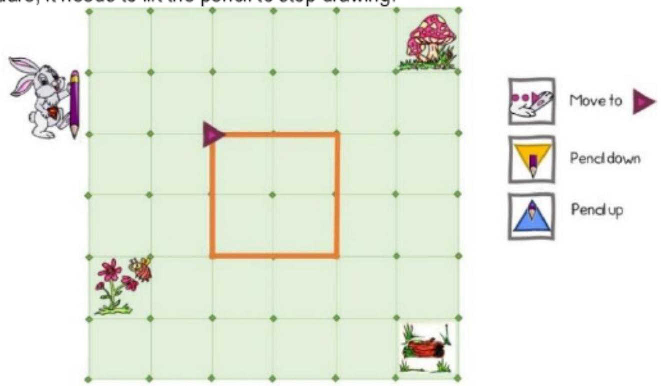
3.4.1 Term 1

Content (Grade 3 / Term 1)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	
Example: A rainbow parrot has four chicks  2019-TS-Elementary-Question-Paper.pdf (olympiad.org.za) • Each young parrot has a different colour for each of its 4 body parts. The colours are red, blue, green, and yellow. • None of the parrots have the same colour body parts as any of their brothers. • Based on the first 3 chicks, what will the colour of the 4th chick's tail be?	Learners need to identify the colour pattern and determine what colour the fourth chick's tail should be.
Algorithm Design and Coding	
C.1 Apply computational thinking skills to develop a set of logical instructions to solve a problem.	**Link to C.6**
Example: Read the instructions in the blocks and develop a set of commands for the dog to pick up his bone and then go home.  **Note:** ***Turn Left*** or ***Turn Right*** means turn to face the direction into which you want to walk (not moving forward but stay in the same block if you turn, facing the direction into which you will continue)) ***Forward*** means move one block forward. For the shortest solution, you need the following commands: 10 Forward commands 4 Turn Right commands 1 Turn Left command 1 Pick-up command	Problems can become increasingly more complex by making the grid bigger, adding constraints and obstacles and adding more actions. One can extend the problem by asking learners to indicate which instructions should be added to follow an alternative route. **Note** For Grade 2 and Grade 3, whenever the character/sprite is instructed to turn left or right, the orientation of the front of the character/sprite should be considered. This implies that the character/sprite face the direction in which it will move next (this requirement is not applicable to Grade R and Grade 1)

Content (Grade 3 / Term 1)	Notes/Examples
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands single repetition and conditional constructs.	**Link to C.1**

Example 1: Drawing (Pen based coding - incorporating "pen-based" drawing and coding activities**)**

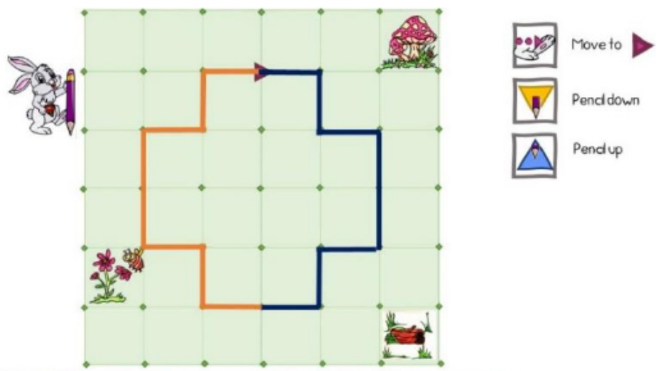
Sipho Super Bunny (SSB) likes to draw. It first needs to move to the starting point on the grid, then put the pencil down to draw a square. After drawing the square, it needs to lift the pencil to stop drawing.

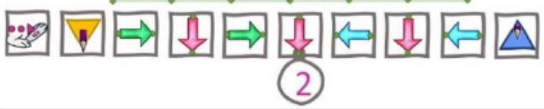


As a variation, the teacher could provide the code which learners need to interpret and evaluate for accuracy.

Example 2 – Drawing and symmetry

In this example, the orange line (left – in front of the arrow) is provided and the learners must develop the code for SSB to draw the symmetrical shape/ mirror image (blue line –on the right right).





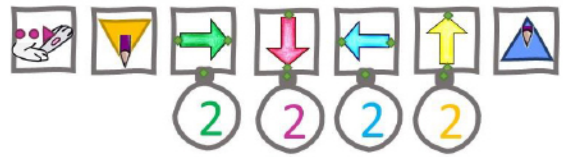
Notes/Examples

New commands are being introduced:

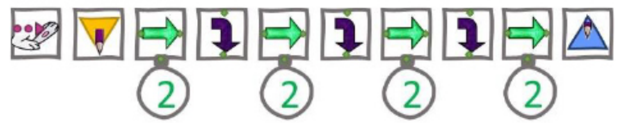
- Move to a starting point on a grid.
- Put down the pencil (to be able to start drawing)
- Lift the pencil up (to stop drawing)

The instructions could be used in various ways, e.g., as the example on the left.

It can then use repetition constructs (indicators) to shorten the number of instructions as indicated below:



Learners could also use *Turn Right* or *Turn Left* commands as an alternative solution:



Learners should be exposed to various alternative solutions to stimulate thinking processes and to compare different solutions in terms of aspects such as efficiency (shortest routes, less instructions, etc.)

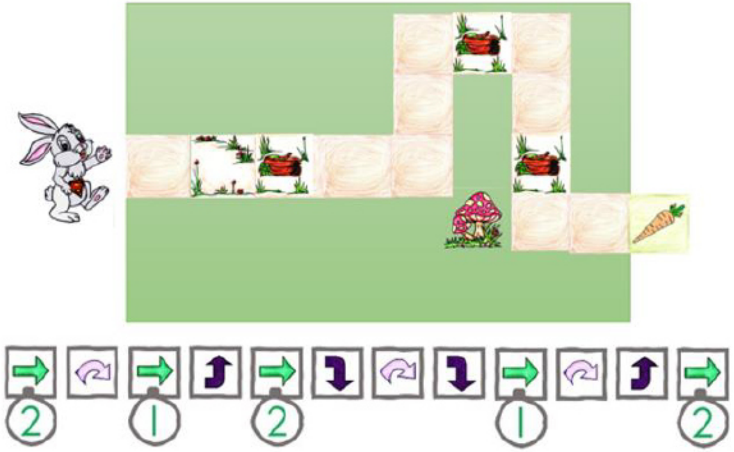
Activities could be adapted to gradually increase the difficulty and complexity levels.

Content (Grade 3 / Term 1)	Notes/Examples
C.3 Interpret and execute a given symbolic or written set of commands	**Link to C.1, C.2**

Example 1- Interpret and follow instructions

SSB can only move one black at a time, jump, turn left and turn right.

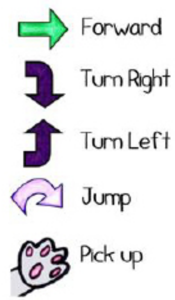
Study the grid below and write down the instructions and how many times SSB must move forward to reach the green tile with the carrot at the end.

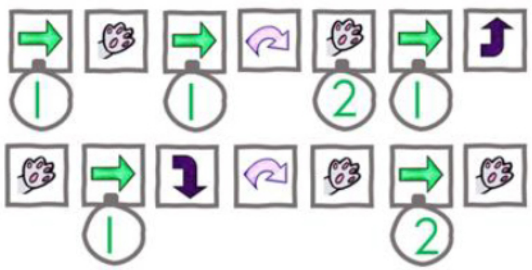


Example 2 – Interpret and execute code

Sipho can only move one block at a time, jump, turn left, turn right and pick up a carrot in a block it resides in.

Study the grid below and write down the instructions and how many times SSB must move or jump, turn left or turn right to get to the green tile





Learners must now execute the code to assess whether it is correct or not.

In the second example, the complexity is being increased (from Grade 2) by adding an additional instruction (pick up)

Evidence suggests that pupils should be taught – initially at least – in small bite-sized chunks. These steps in the learning process should be well-thought out and gradual as well as allow plenty of opportunity for practice (see, for example, Rosenshine, 2012; Coe *et al*., 2014; Sealy, 2019).

Note:

In addition, referring to example 2, the learners can also be expected to present their solution in an algorithmic form using two- or three-word phrases. For example:

Equivalent algorithm

Start
- Forward
- Pick up
- Forward
- Jump
- Pick up x 2
- Forward
- Turn Left
- Pick up
- Forward
- Turn Right
- Jump
- Pickup
- Forward X 2
- Pickup

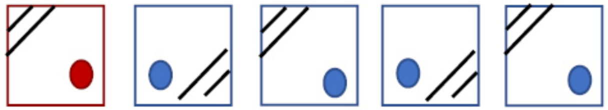
Stop

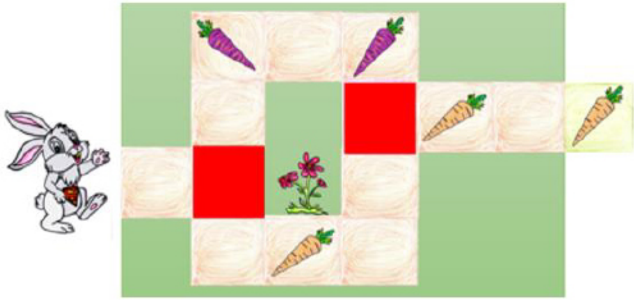
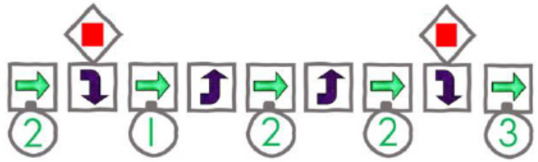
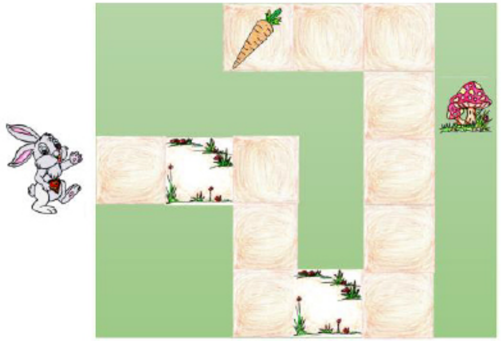
Content (Grade 3 / Term 1)	Notes/Examples
Robotics (± 2 hours)	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	**Link to R.2**
R.2 Identify different types of robots.	**Can be done with R.1**

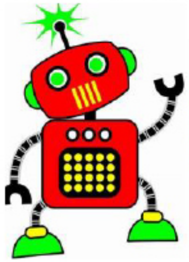
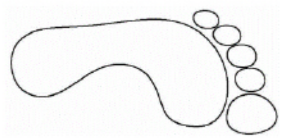
Highlight the following about a robot, using an example:
• Special kind of machine
• Follow instructions to do things by itself (are coded)
• Have different parts that can help them do different things such as
  ○ arms and hands to pick up something.
  ○ sensors to sense (see / hear) what is around them.
  ○ legs or wheels for moving around.
• Made by people to help them do different things such as vacuum the house, assembling cars.
• Do things that people cannot do such as working in dangerous places
	Reinforce and extend from previous Grades

R.5 Design a simple product (artefact) based on a set of design specifications.	**Reinforce and extend**
R.6 Mimic the operations of a robot.	**Reinforce and extend**
Digital Concepts (± 1 hours)	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Link to D.2 and D.3**
Using an appropriate example, explain that technology is designed with a purpose of solving problems that meet human needs and wants. It refers to tools, machines, or devices that make our lives easier or better. **Examples** of technology include computers, smartphones, TVs, video games and even 'robots' that perform specific tasks. Information Technology (IT) is a type of technology that deals with information, such as data, images, and sound. IT includes things like computers, software, and the internet. The **purpose** of IT is to help people access and use information more easily and efficiently. Technology is all around us, and we use it every day to communicate, learn, and have fun.	Learners need to distinguish between technology and information technology and provide some examples and the purpose of each.
D.2 Recognise that he or she is living as citizens in a digital world.	
Learners need to understand that a digital footprint is like a trail of your activity that you leave behind when do something online, e.g., use WhatsApp and that, even if you delete a WhatsApp message, it is very possible that somewhere there remains a record of it. **Possible discussion points** Just like how one leaves footprints in the sand when you walk on sand, you also leave behind traces of your online activity such as the websites you visit, the videos you watch, the pictures you post, and the things you say or search for online. This digital trail can be seen by other people, like your friends or family, and even by people you don't know. It's important to remember that anything you do or say online can be traced back to you, so it's important to be careful about what you share online and to always think before you post	Reinforce from previous Grades and Terms • Digital Footprint Empower learners with thinking skills and strategies to support the development of self-awareness and self-regulation so they learn to manage themselves as digital citizens
D.3 Demonstrate an understanding of the concept of a computing device.	**Reinforce and extend from previous Terms and Grades**
Assessment – Term 1	
Continuous Assessment – Refer to Section 4	

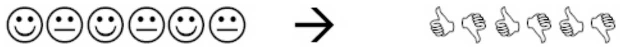
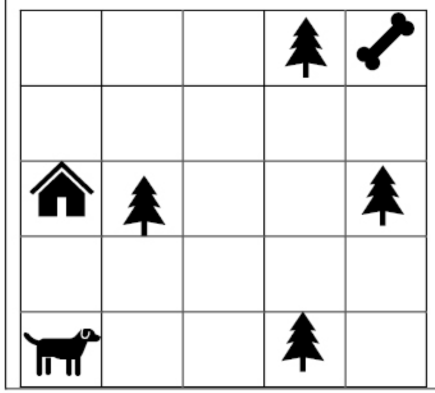
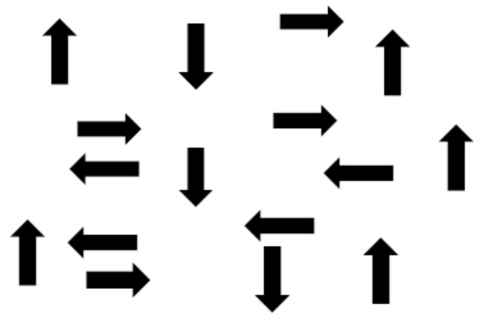
3.4.2 Term 2

Content (Grade 3 / Term 2)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Link to C.1**
Example 1:  Example 1: Different shapes but all have the same number of angles and sides (the pattern) **Example 2**: What comes next? Why? 	Learners must interpret and explain the patterns. Distinguish between repeating and growing patterns
Algorithm Design and Coding	**Link to C.6**
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Link to C.2 and C.3**
Example: 1. Explain to the learner that they will be using a grid to practice sequencing and directional commands, just like a computer programmer. 2. Show the learner the set of directional cards and explain what each card means (e.g. up, down, turn left, turn right, forward) 3. Show the learner the set of number cards and explain that they will be using these cards to move a character on the grid. 4. Place the character (a small sticker or marker) at the bottom-left corner of the grid. 5. Ask the learner to choose a number card (e.g., 3) and a directional card (e.g., up) and place them in a sequence (e.g., 3, up) 6. Ask the learner to follow the sequence and move the character on the grid accordingly (e.g., move up three squares) 7. Repeat steps 5-6 several times with different sequences of numbers and directional commands, encouraging the learner to think carefully about the order of the commands	In this activity, learners eventually design their own 'game' in C.2 This activity helps learners reinforce their sequencing and directional skills, as well basic programming concepts in a fun and engaging way.
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands, single repetition, and conditional constructs.	**Link to C.1 and C.3**
Using the example in C.1, once the learner is comfortable with moving the character on the grid, challenge them to create their own sequences of commands using the number and directional cards	**Extend activity from C.1**
C.3 Interpret and execute a given symbolic or written set of commands.	**Link to R.6**
Example 1: Decision with pair programming SSB does not like purple carrots at all. When SBB lands on a red block, it must decide which of the instructions on the right to follow to avoid collecting purple carrots. Divide learners into pairs and provide each pair with the grid and the instructions (completed code) below.  One learner act as the navigator and the other act as the driver. The navigator reads the instructions provided and the driver act as the rabbit and follow the instructions as communicated by the driver. When done, driver explains the code to navigator and the navigator verifies that the explanation is correct.	The complexity is being increased by adding two decision structures (as one compared to Grade 2) to the problem, combined with several repetition structures.

Content (Grade 3 / Term 2)	Notes/Examples
  **Example 2 Interpret instructions provided and fill in missing instructions.** SSB must collect the carrot. Study the grid and the code instructions below. Interpret the instructions provided and fill in the missing instructions to enable SSB to collect the carrot.  **Forward x3** **Down x2** **___________ (missing instruction(s))** **Up x____ (missing number)** **Left x2** Now code the completed instructions (algorithm) using arrows and numbers	
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	**R.1 and R.2 can be done together**
R.2 Identify different types of robots. **Example** Learners should now have a good idea that a robot is such as body, arms and hands, sensors, control system, power source, wheels or legs, tools or attachments	**Reinforce from previous Graders and Terms using different examples**
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.1 and R.2**
Example: Provide an example of a robot artefact that learners need to design. See R.5 Grade 1, Term 3 Learners get inspiration form their understanding of robots and then ideate about their robot: • Who their robot is. • What type of robot it is • What components it has • What their robot can do • How their robot could help people	Learners should be able to design an artefact that • "looks like" a robot. • can "move like a robot". • have "parts like a robot". follow instructions like a robot

Content (Grade 3 / Term 2)	Notes/Examples
R.6 Mimic the operations of a robot.	**Link to R.5 and C.3**
Use examples of robots (pets, robots, cars) and directional coding cards and let learners create instructions for these robots and act it out.  	Reinforce and extend from previous Grades and Terms using different activities
Digital Concepts (± 1 hours)	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Reinforce from previous Grades and Terms**
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1**
Example: Positive Footprint Provide learners with an 'empty' footprint. Let them write down in the footprint what they would like other people to link to their name in future, thinking about aspects such as: Their profile, Achievements, Pictures they shared, how they behaved online 	Reinforce from previous Grades and Terms • Digital Footprint
D.3 Demonstrate an understanding of the concept of a computing device.	Learners need to understand that computing devices are all around us and that these are all linked to facilitate, e.g., communication
Possible Discussions What is a computing device? How we can use computing devices, e.g., computing devices help us to communicate and learn How do we need to behave when we use computing devices	
Assessment – Term 2	
Continuous Assessment – Refer to Section 4	

3.4.3 Term 3

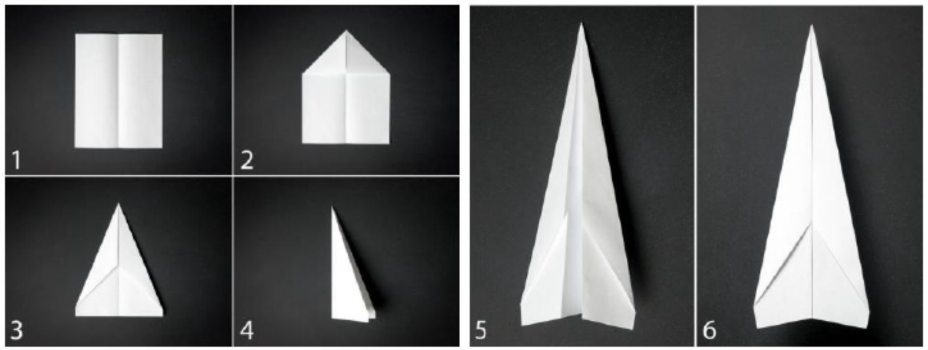
Content (Grade 3 / Term 3)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**C.6 and C.7 can be done together**
Example 1 Interpret the pattern, then translate pattern using smileys to pattern using thumbs. 	Interpret a pattern and then translate the pattern (create the same pattern using different symbols/material)
C.7 Create or complete a pattern to represent a data set. **Example 1** Growing patterns: □ □□ □□□ □□□□ ___ and their underlying rule (+□ in the above example) **Example 2** Decreasing growing patterns and their underlying rule ((-□ in the example below) □□□□ □□□ □□ ___ **Example 2** Determine the pattern (rule), then complete the pattern.  Learners must first determine the underlying rule and then complete the pattern accordingly.	Extend patterns to include growing patterns and identifying the underlying rule. Learners can also play games with repeating as well as growing patterns, such as a domino game in which they need to match patterns with the same unit or rule.
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**C.1, C.2, C.3 and C.4 can be done together**
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands single repetition and conditional constructs. **C.3 Interpret and execute a given symbolic or written set of commands** **Example**: Moving one block at a time, the dog must first pick up the bone and then go to its kennel. Use the arrows provided, avoiding the trees, pack out the route to the bone and then going to the kennel.  	In the example on the left, learners must solve the problem using • C.1 – apply computational thinking to develop a solution (algorithm) • C.2 – present a solution using arrows • C.3 – act out (implement) the solution and test if it works • C.4 – find the error and correct it (debug) if it does not work As there is more than one route to the bone and back, one can also extend the problem by asking learners to indicate how many routes the dog can follow, and which route is the shortest.

Content (Grade 3 / Term 3)	Notes/Examples

C.4 Debug a given symbolic or written set of instructions.

Example 1

Study the following picture steps and description to fold a basic paper plane.
(https://www.hgtv.com/design/make-and-celebrate/handmade/how-to-make-a-paper-airplane)



Steps

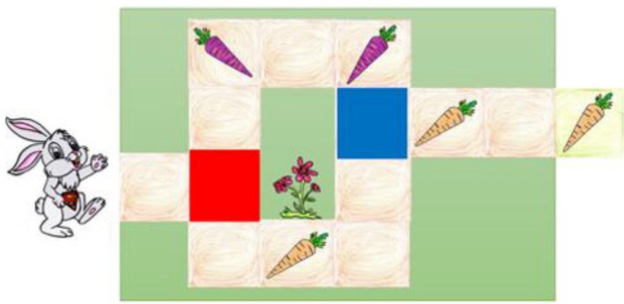
1. Unfold the paper and fold each of the top corners into the center line.
2. Fold the paper in half vertically.
3. Fold the plane in half toward you.
4. Fold the wings down, matching the top edges up with the bottom edge of the body.
5. Fold the top edges into the center line.
6. Add double stick tape to the inside of the body. The finished plane should look like this.

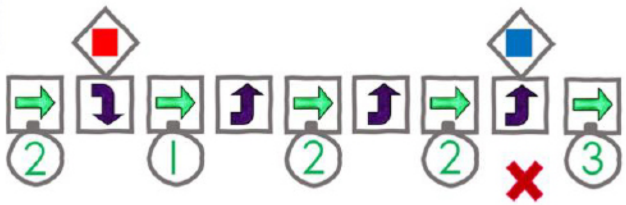
Fold your own paper plane looking at the picture steps and the written steps.

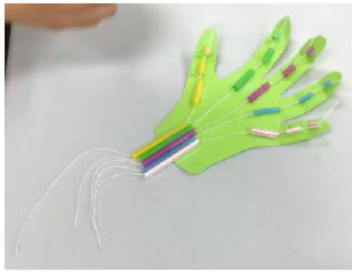
How should the written steps be renumbered to match the steps as per the pictures?

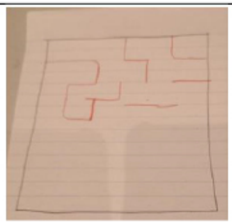
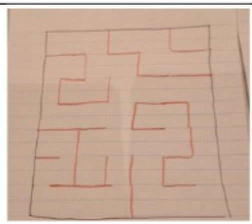
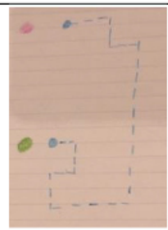
Example 2:

SSB only like arrange carrots. Which path should he follow to eat only the orange carrots. There is an error in the solution on the right, Find the error and correct.





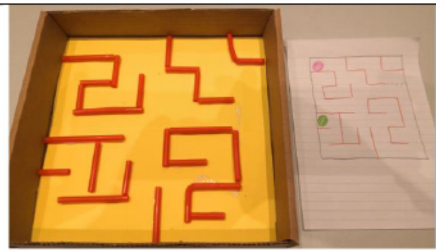
Content (Grade 3 / Term 3)	Notes/Examples
Robotics	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	Reinforce and extend from previous Grades and Terms using different examples and activities.
R.2 Identify different types of robots.	
R.3 Outline the different components of a robot	
R.4 Present an understanding of how robots affect the world.	
Assisting People: Robots can be programmed to assist people who may need help. Some robots can help people with disabilities by fetching things, opening doors, etc. **Exploring New Places**: Robots are also used to explore places that are hard for humans to reach. E.g. they can go deep into the oceans to study sea creatures. **Learning and Education**: Robots can be great tools for learning. Some schools use robots to teach students about science, technology, engineering, and math	
R.5 Design a simple product (artefact) based on a set of design specifications.	**Link to R.1, R.2, R.3 and R.4 and includes R.5, R.6, R.7**
Example 1 Design a robot hand!   https://www.youtube.com/watch?v=e1c095iTIqs	Refer to design process in Grade 1 Term 3 to first design the hand.
R.6 Mimic the operations of a robot.	**Link to R.5**
In relation to the robotic hand artefact the students can design their own simple sign language to communicate messages pulling the strings of the hand lifting certain fingers. An operation to turn the hand around can also be included to either have the fingers pointing down or up. This extends the range of the symbols that can be presented.	Reinforce from previous Terms and Grades
Marble maze: The learners design a simple maze on paper. Through which a marble can run when tilting. Plan the design.	**Marble maze** Blocks paper may be used for this activity. This example covers several coding and robotics skills including elements of the design thinking process. The debugging process is also applied in various forms firstly through the checking of the maze and secondly through the checking of the algorithm to move the marble from point A to B. Various fine motor skills are also developed using this activity/

Start planning the design. Rethink if needed	Finalise the design	Indicate starting point (green) and end point (red)	Test the design (on paper with an overlay) indicating the path the marble will follow
			

<table>
<tr><th>Content (Grade 3 / Term 3)</th><th>Notes/Examples</th></tr>
<tr><td>

Implement the design (build the artefact)

Create the solution to move the marble from the start to the end following the desired path.

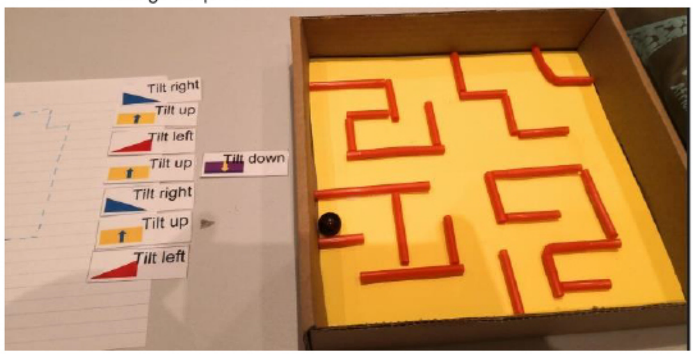
<table>
<tr><th>Material</th><th>Maze</th><th>Test against overlay</th></tr>
<tr><td>Use cardboard or paper and a recycled box</td><td>Cut straws (single and double length straws that can be done in advance) and pack straws onto cardboard to represent the maze.</td><td>Use overlay to see if maze is correct.</td></tr>
<tr><td></td><td> Check layout against design.
If correct, glue straws to cardboard</td><td></td></tr>
</table>

Design coding cards (can be done in advance) – could be substituted with written instructions:

Tilt Right, Tilt left, Tilt up, Tilt down

Code the solution (Design the algorithm)

- The learners must now develop the algorithm (solution) to have a marble move from the indicated starting point (green) to the end point red).
- Compare the algorithm to the original paper-based (overlay) solution to that of the coding cards.
- Test the solution by following the code (card instructions) and tilt the box according to the instructions.
- Debug if required.



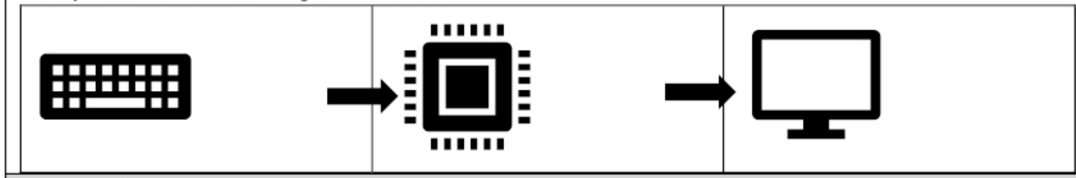
Note:
This process may require further debugging.

</td><td>

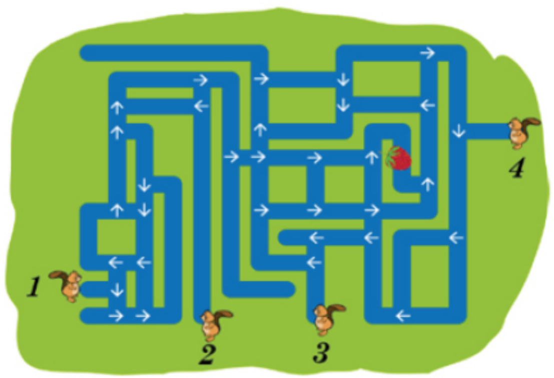
Learners could then exchange mazes and solve each other's maze problems, first using coding cards and then physically executing the code.

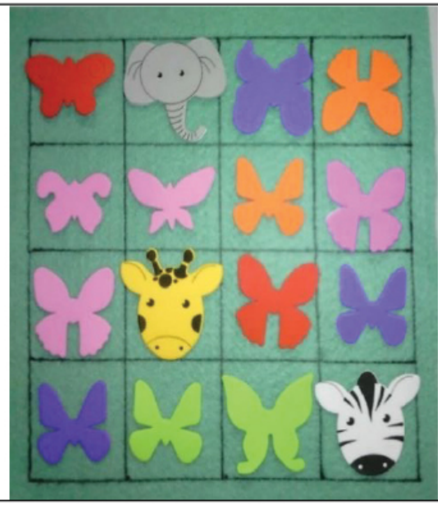
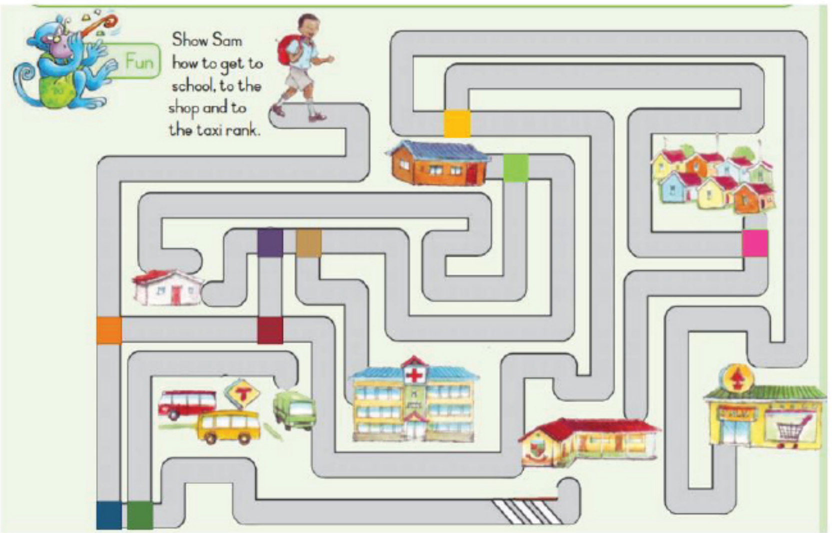
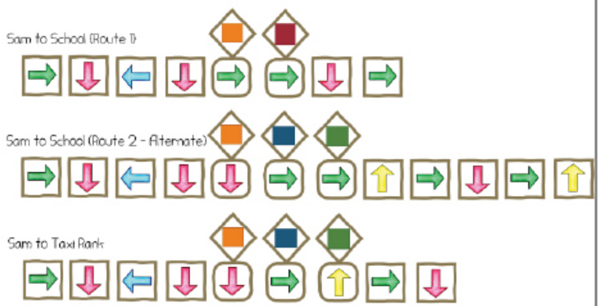
The solving of the physical game requires a certain degree of hand-eye coordination.

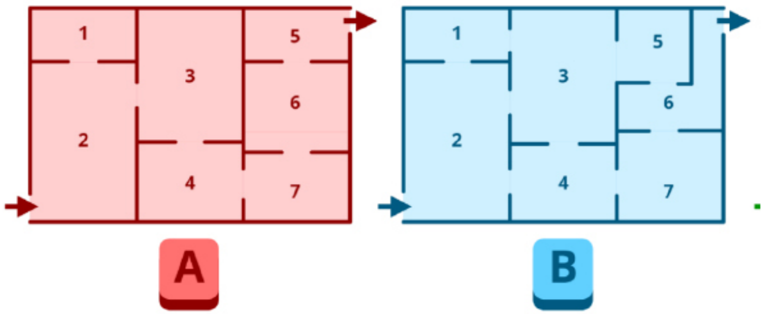
</td></tr>
<tr><th>Digital Concepts</th><th>Could be integrated with Life Skills or Language</th></tr>
<tr><td>D.1 Outline the concept of technology and purpose of information technology (IT).</td><td>Reinforce and extend from previous Grades and Terms using different examples and activities.</td></tr>
<tr><td>D.2 Recognise that he or she is living as citizens in a digital world.</td><td>Link to D.3, D.4</td></tr>
</table>

Content (Grade 3 / Term 3)	Notes/Examples
Example Digital Citizenship discussions: Prepare cards with different actions or choices children make online, like "sharing a funny video," "commenting on someone's post," or "playing a game with strangers", etc. Divide learners into small groups and have each group draw a card. Each group discusses how their actions can leave a "digital footprint or could hurt someone's feelings, etc. Talk about the positive and negative consequences of different actions they take online and highlight the values that must be instilled and the character we should express.	Reinforce and extend using different examples and activities As digital citizens, we need instil positive values that build character to guide our behaviour when we interact in the digital world. Learners need to start to understand the impact of their actions in the digital world and develop essential skills for being responsible and respectful digital citizens.
D.3 Demonstrate an understanding of the concept of a computing device.	**Link to D.7**
Learners need to understand that a computing device is a machine that can process and store data and information, and can be used to do things like play games, watch videos, and communicate with others	
D.4 Identify the common uses of ICT in the real world.	**Link to D.5**
Communication technology plays a crucial role in an ICT system ICT is an umbrella term that includes any communication device, encompassing radio, television, cell phones, computer and network hardware, satellite systems, and so on, as well as the various services and appliances with them such as video conferencing and distance learning Communication technology enables modern computing and allows people and organizations to interact in the digital world	Reinforce and extend from previous Grades and terms using different examples and activities.
D.5 Differentiate between the components of an ICT system.	**Link to D.1, D.3, D.4, D.5 and D.7**
Possible discussion An ICT system is a system that uses computers, programs (instructions), and networks to process and store information. It has distinct parts that work together to help us use computers to do different tasks: • The first part is the computer itself. • The second part is the instructions (software) that tell the computer what to do. • The third part is the data. Data is the information that we put into the computer to work with. This can be things like pictures, music, or text. • An ICT system also has networks. Networks are like the roads that help computers talk to each other. They connect computers together so that we can share information and work on projects together.	Reinforce and extend using different age appropriate examples and activities. Use pictures to discuss the different components of an ICT system.
D.7 Present a basic understanding of the concept of input processing and output.	**Link to D.3 and D.8**
Learners need to understand that a computing device receives input, processes the input and provide output	Reinforce and extend from previous Grades and terms using different examples and activities.
D.8 Interpret a pattern to represent or communicate a message or image.	**Link to D.3 and D.7**
Example: Wat does the following communicate? 	One could communicate input, processing and output using symbols, e.g., input (keyboard), processing (CPU/chip) and output, e.g., computer screen.
Assessment – Term 3	
Continuous Assessment – Refer to Section 4	

3.4.4 Term 4

Content (Grade 3 / Term 4)	Notes/Examples
Pattern Recognition	**Could be integrated with Mathematics or Language**
C.6 Recognise and interpret patterns in symbolic sets of data or visualisations.	**Also link to C.7 and C.1**
Example Study at the following text: A A B A A C A A D A A B A A B A Find the following pattern within the above text: A A B A Pattern found in three instances at positions 0, 9 and 12: A A B A / A A B A / A A B A A A B A A C A A D A A B A A B A 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 One could start with asking learners to first find the BA patter (position 2, 11 and 14)	Find hidden patterns / patterns within data Pattern recognition eventually leads to analysing patterns in data. By identifying patterns, we can predict what will come next and what will happen again and again in the same way. A pattern may be numerical, visual or behavioural. In Computer Science/coding we analyse patterns in data and make predictions and generalisations based on the pattern analysis.
C.7 Create or complete a pattern to represent a data set.	**Link to C.6 and C.1 and C.2**
Example (i)  ________ (ii) tttt, TTTT, ttt, TTT, tt, TT ______ (iii)  ________ (iv) 4, 8, 12, 16, 20, ______, ______	Provide learners with patterns they need to complete. Learners also need to create their own patterns. In programming, for example, a program that can recognize patterns in handwriting can be used to digitize handwritten documents, while a program that can recognize patterns in speech can be used to transcribe spoken words into text. Pattern recognition is also a fundamental aspect of artificial intelligence and plays a key role in many applications of machine learning,
Algorithm Design and Coding	
C.1 Apply computational thinking skills do develop a set of logical instructions to solve a problem.	**Link to C.2**
Example Four beavers start swimming from different places. They only swim forwards and always follow the arrows. Select all the beavers who will reach the strawberry. [TS-2018-Solutions-Guide.pdf (olympiad.org.za)] 	Use abstraction to highlight the vital information for solving the problem and ignore unimportant information. Decompose by follow instructions (algorithm) for beaver 1, then for beaver 2, then for beaver 3 and lastly for beaver 4 (one at a time) to see which beavers will reach the strawberry

Content (Grade 3 / Term 4)	Notes/Examples
C.2 Present a simple coding solution using symbolic or written statements representing sequences of commands single repetition and conditional constructs.	**Reinforce and extend from previous grades and tems**
Example Using the following as an example question may be posed such as - Can Zebra visit elephant without crossing any pink butterflies? - Can Giraffe visit Zebra by crossing every single pink butterfly? Which route should Elephant follow to cross as many as possible different colours of butterflies? (Elephant may only cross a single colour butterfly once). 	Provide learners with a problem and let them develop the instructions to solve the problem, then present and execute the set of instructions. Additional complexity which requires the application of analysis and simple problem-solving skills could be added to problems.
C.3 Interpret and execute a given symbolic or written set of commands	
Example The following example is from the DBE Rainbow Grade 1 (English HL – Book 1, page 115). With simple adaptations it can easily be changed to a problem-solving question that require some coding and decision making. The coloured squares indicate conditional stops.   	Learners develop different routes for Sam to go to school or to go to the bank.

Content (Grade 3 / Term 4)	Notes/Examples
C.4 Debug a given symbolic or written set of instructions.	Debug your algorithm, if necessary
C.5 Evaluate a given solution towards potential improvement.	Learners can compare algorithms in pairs to decide whose algorithm is more effective or how it can be improved, e.g., find pattern / combine single steps into a loop.
Example: Museum visit (extended from Grade 2) Visitors visiting the museum are only allowed to go through all the rooms exactly once. This is called a one-way tour. Therefore, the following restrictions apply: **Restrictions** The may not visit a room more than once. They are also not allowed to use the same door for entering and exiting a room. The visitors must start at the arrow that enters the museum and leave by way of the door with the arrow leaving the museum. **Task** Evaluate the following floor plans to see which museum layout will meet the requirements (restrictions)  Adapted from 2021-TS-Elementary-Question-Paper.pdf (olympiad.org.za)	Complexity is increased by adding rules or restrictions
Robotics (± 2 hours)	**Could be integrated with Creative Arts or Language**
R.1 Explain what a robot is in simple terms.	Learners should now be able to • Describe a robot. • Identify types of robots. • List the various parts that a robot could have. • State what a robot can do (and cannot do). • Describe how robots move. • Explain how robots impact the world around us.
R.2 Identify different types of robots.	
R.3 Outline the different components of a robot	
Components such as body, arms and hands, sensors, control system, power source, wheels or legs, tools or attachments (reinforce from Grade 2 and previous terms) robots-activity-sheets.pdf (ieee.org)	
R.4 Present an understanding of how robots affect the world.	
R.5 Design a simple product (artefact) based on a set of design specifications.	Learners should no be able to design an artefact that • "looks like" a robot • can "move like a robot". • have "parts like a robot" or • follow instructions like a robot
R.6 Mimic the operations of a robot.	Learners should be able to act out a set of instructions (provided or developed) that mimics various actions/tasks a robot could perform.

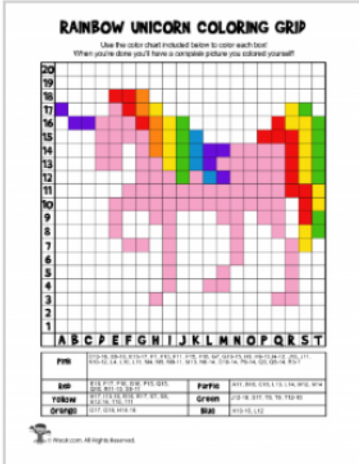
Content (Grade 3 / Term 4)	Notes/Examples
R.7 Create test and execute a set of robotic instructions.	**Links to R.1, R.2, R.3, R.4, R.5, R.6**
Example A learner/group of learners are provided with a grid with obstacles and limitations as well as an outcome (e.g., robot must pick up 5 coins, avoid, x and y and may not touch z and my not step on the same block twice. Learner/group then develop the instructions to solve the problem/meet the outcome. One learner act as a robot and executes (act out) the instructions developed to see if it works. If there is a mistake, the next learner/group of learners must debug and correct, then test again… repeat until it is correct	Learners should create their own set of instructions for a robot to act upon, execute it by acting it out / testing it and debugging it if necessary.
Digital Concepts	**Could be integrated with Life Skills or Language**
D.1 Outline the concept of technology and purpose of information technology (IT).	**Reinforce and extend**
Technology is designed with a purpose of solving problems that meet human needs and wants. It refers to tools, machines, or devices that make our lives easier or better. **Examples** of technology include computers, smartphones, TVs, video games and even 'robots' that perform specific tasks. Information Technology (IT) is a type of technology that deals with information, such as data, images, and sound. IT includes things like computers, software, and the internet. The **purpose** of IT is to help people access and use information more easily and efficiently. Technology is all around us, and we use it every day to communicate, learn, and have fun.	Reinforce from previous grades and terms. Learners need to distinguish between technology in general and information technology and provide some examples and the purpose of each.
D.2 Recognise that he or she is living as citizens in a digital world.	**Link to D.1**
Reinforce from previous grades and terms. We are surrounded by technology, and we use it in many aspects of our daily lives. Provide learners with examples. As citizens in a digital world, we have certain responsibilities and expectations regarding aspects such as online safety, privacy, respect, digital footprint. It's important to remember that anything you do or say online can be traced back to you, so it's important to be careful how you behave online, about what you share online and to always think before you post.	Learners need to understand the following concepts and also provide examples: • Online safety • Privacy • Respect for others • Digital Footprint
D.3 Demonstrate an understanding of the concept of a computing device.	**Link to D.1 and D.2**
A computing device is a machine that helps us process and store information. It can be anything that uses a computer chip to work, like a desktop computer, a laptop, a tablet, a smart phone or a Smart TV. Computing devices help us learn, communicate, create, and have fun. How to take care of computing devices and how to use them safely.	Reinforce and extend from previous grades and terms. Learners need to understand that computing devices are all around us and that a computing device accepts input, processes the input, and provides output. Use appropriate examples.
D.4 Identify the common uses of ICT in the real world.	**Link to D.1, D.2 and D.3**
Common uses include communication, education, entertainment, business. Discuss uses within these areas known to learners	Learners need to identify uses of ICT in real life known to them, e.g., point of sales system at the shop
D.5 Differentiate between the components of an ICT system.	**Links to D.4**
Reinforce and extend from previous grades and terms. **Example**: the sales point in the shop has a scanner that reads the barcode on the item and adda the price of each item to give you the total amount payable. Another part, the card machine reads your banking details and make a payment. (use examples that learners understand) Parts include: • Hardware (input and output devices), e.g. till, barcode reader and the card reader • Software (code) – programs that enable the system to work • Data that is processed and stored, e.g. read barcode on items to get prices and calculate amount due • The Internet (network) that communicates with the bank to make a payment / communication between till and barcode reader or the card reader • People that operate the devices and users that communicate with others using ICT systems	Learners need to understand that an ICT system is made up of various parts or components that work together to help us process and share information Provide appropriate examples (e.g. point-of-sales system) that learners can understand

<table>
<tr><th>Content (Grade 3 / Term 4)</th><th>Notes/Examples</th></tr>
<tr><td>D.6 Explain how the adaption of technology impacted the world we work and live in.</td><td></td></tr>
<tr><td>Reinforce and extend from previous grades and terms.
Impact: (different sectors)
Communication: Made the way we communicate much faster and easier (Instant messaging, e-mail, etc.)
Education: Changed the way we learn (online resources)
Entertainment: Stream movies, etc.
Work: People work remotely, collaborate with people around the world, etc.</td><td>Learners need to understand that technology has transformed the way we work, communicate, and live.
Provide appropriate examples</td></tr>
<tr><td>D.7 Present a basic understanding of the concept of input processing and output.</td><td>Reinforce and extend</td></tr>
<tr><td>Reinforce and extend from previous grades and terms.
Example: Using the point of sales ICT system – the barcode reader inputs the item codes, then process the item prices and then provides output in the form of the amount payable. The person that operates the pay point is also part of the ICT system; so is the code (software instructions) that calculates the prices.</td><td>Learners need to understand, using an example familiar to them, the concepts of input that results in output through processing.</td></tr>
<tr><td>D.8 Interpret a pattern to represent or communicate a message or image.</td><td>Link to D7 and D.9</td></tr>
<tr><td>Example 1.
Yellow and Black stripes: depending on how it is presented it could be interpreted as
Bee: the patter of yellow and black stripes could represent a bee
Warning: the yellow and black strips are commonly used as a waring sign such as on construction sites or traffic cones
Sport team's shirts: A sport team could have shirts with yellow and black stripes.
Example 2 Secret Code
Riddle Code Breaker
Use the code to find the answers to the riddles
<table>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>G</td><td>H</td><td>I</td><td>J</td><td>K</td><td>L</td><td>M</td></tr>
<tr><td>14</td><td>26</td><td>11</td><td>20</td><td>4</td><td>19</td><td>12</td><td>24</td><td>1</td><td>25</td><td>6</td><td>23</td><td>5</td></tr>
<tr><td>N</td><td>O</td><td>P</td><td>Q</td><td>R</td><td>S</td><td>T</td><td>U</td><td>V</td><td>W</td><td>X</td><td>Y</td><td>Z</td></tr>
<tr><td>13</td><td>7</td><td>15</td><td>2</td><td>17</td><td>21</td><td>10</td><td>22</td><td>8</td><td>16</td><td>3</td><td>18</td><td>9</td></tr>
</table>
1. What can you catch but not throw?
14   11 7 23 20

2. What begins with T, finishes with T, and has T in it?
14   10 4 14 15 7 10

1. A Cold
2. A teapot

https://content.twinkl.co.uk/resource/92/4a/za-hl-495-riddle-code-breaker_ver_2.pd</td><td>Revise and extend using different, appropriate examples
An activity such as Example 2 can also be used to illustrate Input, Processing and Output:
Input: Prepare a simple cipher code (secret code) where symbols/numbers replace letters of the alphabet. Write a short message in the code as the "input" for the computer.
Processing: Provide the learner with a decoder chart to figure out the symbols and translate the message. This represents the computer processing the encrypted/coded data.
Output Once the message is decoded, the child can read it out loud as the computer's "output."</td></tr>
<tr><td>D.9 Create a pattern to represent or communicate a message or image.</td><td>Link to D.8</td></tr>
</table>

Content (Grade 3 / Term 4)	Notes/Examples
 Secret message coding bracelet using binary coding with the alphabet to represent message, e.g., the learner's name **Note**: each letter is represented by a sequence of 8 zeros and ones (ASCII code) (Three colours of beads required: one colour to represent 0s and one colour to represent 1s and another colour to represent a symbol such as a heart, star, etc. that will separate the starting and ending points of the letters in the secret code.) **Example 2**  https://www.woojr.com/mystery-picture-grid-coloring-pages-fantasy-fairy-tales/	ASCII code is a set of digital codes (binary numbers) representing letters, numerals, and other symbols, widely used as a standard format in the transfer of text between computers)
Assessment – Term 4	
Continuous Assessment – Refer to Section 4	

Note:

In terms of coding, typically, problems could require learners to

- read code and explain what it does or
- work through (trace) / act out code (physically or simulated) to determine the output or the correctness or
- provide missing code instructions (code instructions are provided with some instructions or code elements missing) that learners need to complete or
- translate verbal/written instructions (algorithm) to code.
- add some functionality/instructions to an existing program.
- rewrite a set of coding instructions to be more efficient, e.g. using a loop construct for code that is repeated or
- choose the correct solution from 2-3 options or
- compare different solutions to evaluate efficiency or
- debug an algorithm or program (find the bug, describe the bug and correct it)
- develop a solution/algorithm (code instructions) based on a given problem or for an open-ended problem through planning, implementing, testing and debugging.

depending on the competency/(ies) the learner needs to demonstrate.

The following example illustrates how a pen-and-paper activity could lead to implement the algorithm in a coding environment.

In Grade 3, the learner could do the pen-and-paper activity only. Implementing the pen-and-paper algorithm in a coding environment will follow in subsequent grades.

The purpose of the example is to illustrate how the unplugged activities done in Foundation Phase, lead to activities that could be done using programming environments.

Example of Algorithm Design (preciseness and detail of instructions) using pattern recognition and evaluation of solution:

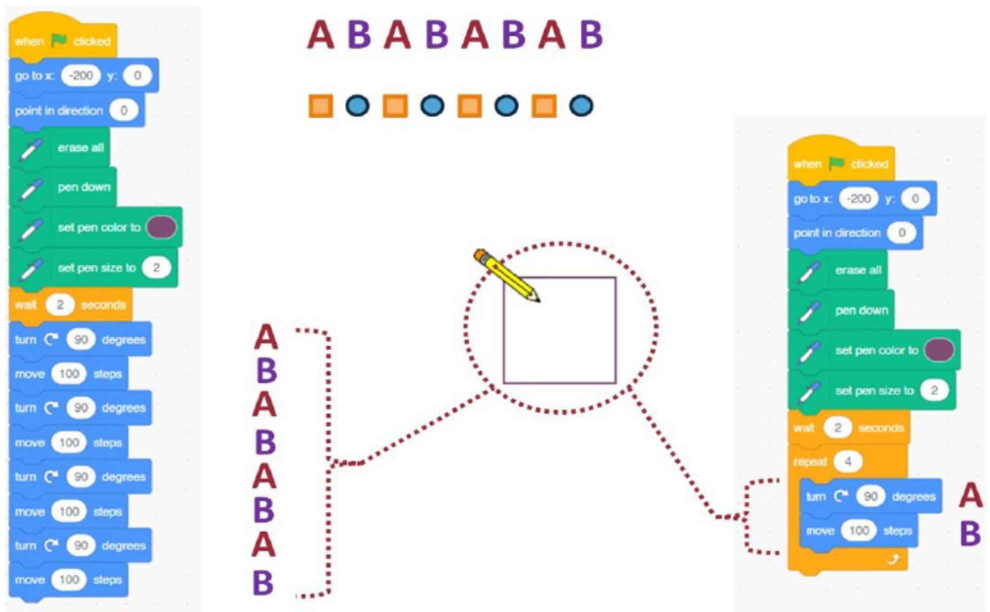
Here is an algorithm – follow exactly (do not look at another's drawings and do not ask for help Step 1: Draw a 3 cm line. Step 2: Draw another 3 cm line. Step 3: Draw another 3 cm line that connects with the line in step 2. Step 4: Draw another 3 cm line that connects to the first line	After drawing, compare your drawing with the drawing of the others. It was supposed to be a square. Now, let us look how to develop a better algorithm to draw a square: Are they different? Why are they different? What was missing from the instructions? It was supposed to be a square.	**Solution 1:** Write down a more precise, detailed set of instructions: 1. Turn right. 2. Draw a 3 cm line. 3. Turn right. 4. Draw a 3 cm line. 5. Turn right. 6. Draw a 3 cm line. 7. Turn right. 8. Draw a 3 cm line.	Now, look for patterns: 1. Turn right. 2. Draw a 3 cm line. [1] 3. Turn right. 4. Draw a 3 cm line. [2] 5. Turn right. 6. Draw a 3 cm line. [3] 7. Turn right. 8. Draw a 3 cm line. [4]	**Solution 2:** Repeat **4** times Turn right. Draw a 3 cm line

Example of two algorithms for drawing a square, the one on the left with linear steps and the one on the right with a loop after identifying repetitive pattern.

Code, Implement, Compare and Evaluate solutions (this could be done in the next phases):

Implement and test both solutions. Both solutions work, however, evaluation shows that the solution on the right is more effective (using a loop) than the one on the right.

The example below shows how the above algorithms can be implemented using programming software:



Note:
Learners need to understand that *turn right* means 90 degrees in Scratch.

Note:
Competencies covered in the above solution:
C.1, C.2, C.3, C,4. C.5 and C.6

4 SECTION 4 ASSESSMENT

4.1 INTRODUCTION

Assessment is a continuous planned process of identifying, gathering and interpreting information about the performance of learners, using various forms of assessment. It involves four steps:

- generating and collecting evidence of achievement.
- evaluating this evidence.
- recording the findings and
- using this information

to understand and thereby assist the learner's development to improve the process of learning and teaching.

In Foundation Phase, all assessment is school-based assessment (SBA) and involves activities that are undertaken throughout the year. Assessment in Coding and Robotics should encourage:

- Computational thinking practices, i.e., integrating the power of human thinking with the capabilities of digital technologies and computer programming.
- Design thinking and design process.
- Problem solving strategies.

In the Foundation Phase, the main techniques of formal and informal assessment are observation by the teacher, oral discussions, practical demonstrations and written recording. Grade R assessment should be mainly oral and practical.

4.2 ASSESSMENT

Assessment is the process of continuously collecting information on a learner's achievement. It is a daily monitoring of learners' progress.

The forms of assessment used should be age and developmental level appropriate. All assessments must cater for a range of cognitive levels and abilities of learners. The design of these tasks should cover the content of the subject in a variety of ways. A variety of forms of assessment (observation, oral, practical and written) should be used to give each learner the opportunity to demonstrate what he or she can do.

However, cognisance should also be taken of what is being assessed. Certain competencies are best assessed with particular forms of assessment. Different kinds of assessments are appropriate to the competencies necessary for different topics at different age groups. It is useful to use an observation checklist to assess learners measuring in the early grades. Rubrics, for example, can be used to evaluate learner's coding and robotics as well as problem solving skills.

Assessment can use the following strategies:

4.3 PROBLEM-BASED LEARNING

Assessment in Coding and Robotics can be done assessing the learner in action, for example, watching the learner solving the problem without stopping the moment. This can be done using the following strategies: As some Coding and Robotics competencies can be integrated with other subjects in Foundation Phase, for example, pattern recognition, it is possible to assess these competencies during the integration process.

4.3.1 Individual Problem-based Learning (coding)

Problem solving is the process of designing, evaluating, and implementing a strategy to answer question, complete a task or achieve a desired goal.

4.3.1.1 *Types of problems*

In terms of coding, typically, problems could require learners to

- provide missing code instructions (code instructions are provided with some instructions or code elements missing / to be completed or
- choose the correct solution from 2-3 options or
- work through (trace) / act out code to determine if it is correct and correct if required or
- rewrite a set of coding instructions to be more efficient or
- compare different solutions to evaluate efficiency or
- translate verbal/written instructions to code (e.g. packing arrows)
- develop the solution/algorithm (code instructions) themselves using computational thinking and following problem-solving process.

The above will depend on the competency the learner needs to demonstrate. Coding problems need to gradually increase in terms of complexity.

4.3.1.2 *Assessing problem-based learning (coding)*

The learner is assigned a problem he/she must solve and in doing so

- needs to understand the problem.
- analyses the problem (what is given and what is needed / what is important and what can be ignored - abstraction).
- identifies the main steps (abstraction / high level solution).
- identifies the detailed steps (decomposition / breaking down the main steps).
- Identifies patterns to determine the need for using coding structures such as repetition.
- implements and tests the solution (algorithm).
- debugs the solution if required.

Refer to Annexure A for rubric example to assess problem solving.

4.3.2 Cooperative Learning

Instead of encouraging learners to compete for grades or achievement, cooperative learning asks them to work together and participate in group learning activities (small groups, e.g. 4 learners), under the guidance of a teacher.

Assessing cooperative learning in Foundation Phase Coding and Robotics

Example rubric to assess cooperative learning activity: *Defining a robot and its different parts.*

Refer to Section 2.6.2.1 for example cooperative learning activity.

Refer to Annexure A for rubric example to assess cooperative learning.

4.3.3 Pair Programming

Assessing pair programming in Foundation Phase Coding and Robotics

Example rubric to assess cooperative learning activity:

Identifying, completing and creating patterns.

Refer to Section 2.6.2.2 for example pair programming learning activity.

Refer to Annexure A for rubric example to assess pair programming.

4.4 Recording and Reporting

Recording is a process in which the teacher documents the level of a learner's performance in a specific assessment task. It indicates learner progress towards the achievement of the knowledge as prescribed in the

Curriculum and Assessment Policy Statements. Records of learner performance should provide evidence of the learner's conceptual progression within a grade and her / his readiness to progress or being promoted to the next grade. Records of learner performance should also be used to verify the progress made by teachers and learners in the teaching and learning process.

Reporting is a process of communicating learner performance to learners, parents, schools, and other stakeholders. Learner performance can be reported in several ways. These include report cards, parents' meetings, school visitation days, parent-teacher conferences, phone calls, letters, class or school newsletters, etc. Teachers in all grades report in percentages against the subject. The various achievement levels and their corresponding percentage bands are as shown in the Table below.

RATING CODE	DESCRIPTION OF COMPETENCE	PERCENTAGE
7	Outstanding achievement	80 – 100
6	Meritorious achievement	70 – 79
5	Substantial achievement	60 – 69
4	Adequate achievement	50 – 59
3	Moderate achievement	40 – 49
2	Elementary achievement	30 – 39
1	Not achieved	0 - 29

4.5 General

This document should be read in conjunction with:

- National policy pertaining to the programme and promotion requirements of national Curriculum statement Grades R-12; and
- The policy document, National Protocol for Assessment Grades R-12

ANNEXURE A – TERMINOLOGY

The following tables provide clarity on terminology used in the CAPS

A.1 CODING

Table A.5 Coding - Clarification of concepts and terms

Term/Concept	Explanation
Algorithm	An algorithm is a set of logical instructions/commands that a human or computer can execute to solve a specific problem or accomplish a particular task. It is a computational process that uses a finite number of steps (logical instructions or commands), carried out in a specific sequence to solve a problem.
Coding	**Coding** is the process of writing instructions that a computer can understand and execute These instructions are written in a programming language, which is a set of rules that define how the instructions should be written. The purpose of coding is to create software programs that can perform specific tasks, such as running a website, playing a video game, or analysing data.
Computation	In computing, **computation** refers to any type of arithmetic or non-arithmetic calculation that is well-defined. It can involve mathematical equations, computer algorithms, and other types of calculations.
Computational Thinking	It refers to a problem-solving approach that involves breaking down complex problems into smaller, more manageable parts and using algorithms and logical reasoning to solve them. It involves skills such as abstraction, decomposition, pattern recognition, and algorithmic thinking. It is a way of thinking that is used in computer science, but it can also be applied to other fields. In education, computational thinking is used to teach learners how to think logically and solve problems systematically.
Conditional (choice/ decision) statement	A control structure that selects one alternative from two or more possible execution sequences to be executed
Control statement	A control structure that is used to modify the order in which instructions are executed such as a loop or conditional statement
Event	A signal or notification that something has happened.
Expression	Refers to a combination of one or more values, operators that can be evaluated to produce a result.
Input	In computing, input refers to the data that is entered into a computer system, such as text, images, or sound,
IPO table	Input-Processing-Output table describes the inputs processing and outputs of a program.
Loop statement	A control structure that allows a sequence of instructions to be continually repeated until a certain condition is reached
Operator	**Operators** are symbols or keywords that represent computations or actions performed on operands. Operators include: Arithmetic operators (+, -, x, /, modulo), comparison operators (=, >, <, ≤, ≥, ≠), Boolean operators OR, AND, NOT, string operators for manipulating strings/text (length, concatenate, indexing) Operators provide the building blocks for creating expressions and performing operations
Output	In computing, output refers to the result of the processed data that is presented to the user in a usable format. This can be in the form of text, sound, image, or video.
Processing	In computing, processing refers to the operations performed by the computer to manipulate or analyse the input data.
Program	A **program** is a sequence of instructions that a computer can execute to perform a specific task.
Trace table	In programming, a trace table is a technique used to test an algorithm and predict step by step how the computer will run the algorithm. Statements are executed step by step, and the values of variables change as an assignment statement is executed. A trace table simulates the flow of execution by showing the values of variables at each step of the algorithm. Trace tables are typically used by novice programmers to understand how an algorithm works and to identify errors in the algorithm 2
Variable	In programming, a variable is a named storage location that holds a value or data. Variables are essential for storing and manipulating data in computer programs. The values in variables can change during the execution of a program.

A.2 Robotics

Table A.6 Robotics - Clarification of concepts and terms

Term/Concept	Explanation
Actuator	Refers to a device that converts energy into physical motion, such as rotation or translation. Actuators are often called the muscles of robots, as they enable robots to perform various tasks and interact with the environment
Controller	Refers to a device that commands, directs, and regulates the behaviour of a robotic system. It takes input signals from the robot's sensors, processes them based on programmed instructions, and then sends output signals to the robot's actuators to perform the desired actions.
Microcontroller	Refer to a type of small computer that can control the functions and behaviour of a robotic system. It generally consists of a processor, memory, input/output ports and other peripherals that can be programmed to perform specific tasks. It can receive data from sensors, process it according to the programmed instructions and send commands to actuators.
Robot	A **robot** is a machine that can perform a series of actions automatically, either by being programmed by a computer or by being guided by an external control device.
Sensor	Refers to a device that can measure or detect some physical property of the environment or the robot itself and convert it into an electrical signal. Examples include light sensor, touch sensor, sound sensor, etc.

A.3 Digital Concepts

Table A.7 Digital Concepts - Clarification of concepts and terms

Term/Concept	Explanation
Cipher	A cipher, also known as an encryption algorithm, is a set of well-defined rules used to transform information into a scrambled form, called ciphertext. It is used to encrypt messages so that they can only be read by someone who knows how to decrypt them.
Computing device	A general-purpose machine that can execute instructions for any data processing purpose. A computing device can receive input, do something with the input and provide a result or output.
Data	Raw, unprocessed facts and figures.
Decode	Reconstructing the original **(encoded)** information. It involves taking an encoded representation and converting it back into its original form
Decrypt	The reverse process of encryption, taking ciphertext and using the appropriate key to convert it back into its original, readable plaintext form.
Digital Citizen	A person who uses the Internet and other digital technology to communicate with other and engage in society.
Digital Citizenship	The ability to participate in online society. It includes concepts like respecting others' privacy, avoiding cyberbullying, netiquette, digital health and welfare, ability to assess the credibility and reliability of online information, intellectual property, impact and responsibility of online actions and deeds.
Digital Footprint	The trail of traceable digital activities, actions, contributions, and communications one leaves behind when using the Internet or digital devices.
Encode	Converting information into a specific format (transforming data or messages into another format)
Encrypt	The process of transforming readable data (plaintext) into an unreadable, scrambled form (ciphertext) using a cryptographic algorithm (cipher) and a secret key.
Hardware	The physical building blocks of a computing device or the tangible parts you can see and touch. It includes: • **Central Processing Unit** (CPU): the component responsible for executing instructions. • **Random Access Memory** (RAM): Component for temporary storage of programs and data the computing device is currently working with. • **Storage devices**: E.g. hard drives, solid-state drives (SSDs), for permanent data storage. • **Input devices** such as keyboard, mouse, screen, microphone mouse, used to interact with the computer. • **Output devices** such as screen, speakers, printer, etc., used to display and output information.
Information	Data that has been processed and organised to convey meaning.
Information and Communications Technology (ICT)	ICT is the use of computing and telecommunication technologies, systems, and tools to facilitate the way information is created, collected, processed, transmitted, accessed and stored
Information Technology (IT)	IT refers to the use of computer systems to manage, process, protect, and exchange data and information.
Input	In computing, input refers to the data that is entered into a computer system, such as text, images, or sound.

Output	In computing, output refers to the result of the processed data that is presented to the user in a usable format. This can be in the form of text, sound, image, or video.
Personal information	In computing, **personal information or personal data** is any information that can identify a person, from one's name and address to one's device identifier and account number.
Processing	In computing, processing refers to the operations performed by the computer to manipulate or analyse the input data. This includes executing software applications, performing calculations, sorting and filtering data, and running programs.
Software	The intangible programs and applications (instructions) that give life to the physical components. Examples include: • **Operating System** (OS) that manages the hardware resources and provides a platform for running other programs. (e.g., Windows, macOS, Linux) • **Application software**: Specific programs designed for performing tasks like word processing, image editing, games, etc. • **Programming languages** used to create new software by writing instructions the computer can understand.
Technology	Encompasses any tool, technique, or process used to solve problems and manipulate our environment. Technology is designed with a purpose of solving problems that meet human needs and wants. It refers to tools, machines, or devices that make our lives easier or better.

ANNEXURE B – EXAMPLE RUBRICS

B.1 PROBLEM-SOLVING (CODING)

Example checklist to assess individual problem-based coding activity:

The process of creating a logical set of instructions to solve a problem or that a robot can mimic, which require a deep understanding of computational thinking and the problem-solving process				
Learner can	**Beginning (1)**	**Developing (2)**	**Accomplished (3)**	**Exemplary (4)**
Explain the problem in his/her own words	Learner cannot explain the problem in his/her own words.	Learner attempts to explain the problem in his/her own words but continuously seeks assistance from the teacher.	Learner can explain a problem in his/her own words but leans on peers for support. Some points are vague.	Learner can skilfully explain a problem in his/her own words and can self-correct.
Identify what is given and what is needed	Learner cannot identify what is given and what is needed	Learner hesitantly attempts to identify what is given and what is needed, but still seeks continuous assistance from the teacher.	Learner hesitantly attempts to identify what is given and what is needed. Some points are vague.	Learner confidently identifies what is given and what is needed
Provide the main steps to solve the problem	Learner cannot provide the main steps to solve the problem	Learner attempts to provide the main steps to solve the problem but will continuously refer back to the teacher for confirmation	Learner hesitantly attempts to provide the main steps to solve the problem, but he/she might miss some steps or give a vague explanation of some of the steps.	Learner confidently provides the main steps to solve the problem
Break the main steps into smaller, easier to solve, detailed steps/parts	Learner cannot identify detailed, easier to solve steps	Learner attempts to identify detailed, easier to solve steps, but continuously seeks assistance from the teacher	Learner attempts to identify detailed, easier to solve steps, but some steps may be vague or incomplete. May lean on peers for support	Learner can identify detailed, easier to solve steps. He/she can self-correct
Implement and test the solution	Learner cannot implement and test the solution	Learner attempts to implement and test the solution but cannot follow through	Learner hesitantly attempts to implement and test the solution. May lean on peers for support.	Learner can confidently implement and test the solution
Debug the solution if required (full marks if learner indicated it correct and teacher confirmed correctness)	Learner cannot identify whether the solution requires debugging or not.	Learner attempts to identify whether a solution needs debugging but is not sure or will try to debug because they are not completely sure that the solution is correct.	Learner will be able to identify whether the solution is correct or not but will not be able to debug.	Learner can confidently identify whether the solution is correct or needs debugging and can do the debugging.

As the rubric above uses a 4-level scale, the learner's problem-solving mark (out of 7 – for reporting purposes) from the above rubric can be summarised as follows:

Problem-solving (coding) summary (generic example)	Mark achieved (from rubric above)
Did the learner understand the problem?	2
Did the learner analyse the problem (what is given and what is needed)?	3
Did the learner identify the main steps (abstraction / high level solution)?	2
Did the learner identify the detailed steps (decomposition / breaking down the main steps)?	1
Did the learner implement and test the solution?	3
Did the learner debug the solution if required?	2
TOTAL	**13/24**
%	**54%**
RATING CODE (for reporting): Adequate achievement (50-59%)	**4**

B.2 Cooperative Learning

Example rubric to assess cooperative learning activity: *Defining a robot and its different parts (See section 2.6.1)*.

	Learner name	#Definition of robot	#Flashcards utilised well.	#Drawing illustrates robot	*Learner fulfilled role well
1.					
2.					
3.					
4.					

#Replace with suitable criteria depending on the task/problem
*Will remain the same irrespective of task/problem

Note:
Although all learners in the group get the same mark for the first three criteria, each learner gets an individual mark for the "Learner fulfilled role well" – this is based on how well each learner contributed based on their set role.

The teacher can give mark these while learners are completing the activity and hence it should not require much extra time.

Each of the aspects listed in the table above, could be assessed using the following example:

Aspect assessed	Beginning (1) 	Developing (2) 	Accomplished (3) 	Exemplary (4) 
Definition of concept, e.g. robot	Key information is missing (e.g. no parts included) and the definition is unclear and difficult to follow	Some key information is included, and the definition is generally clear and easy to follow but may be incomplete or somewhat disorganised.	Most of the key information is included (e.g. most of the parts) and it is mostly well-organised and easy to follow	The learner demonstrates full understanding in that the definition is well-organised, complete, and easy to follow.
Flashcard utilised well	Flashcards are not used effectively	Some attempt is made to use the flashcard to explain the concept, but it lacks detail and key information	Flashcards are used appropriately to explain the concept and includes most of the key information	Flashcards used effectively/innovatively to support a complete explanation of the concept and all key information
Drawing illustrates concept, e.g. robot	Drawing attempts to convey the concept, but the drawing is incomplete and/or difficult to interpret	Drawing includes some relevant details that may not all be accurate and conveys the concept but lack detail	Drawing includes most of the relevant and accurate details that appropriately convey the concept	drawing includes rich, and accurate details that effectively convey the concept.
Learner fulfilled role well	Learner does not understand his/her role and makes no contribution or unrelated contributions	Shares ideas or tries to fulfil her/his role, but does not work with group and most of the contributions are unrelated	Tries to understand his/her role and mostly makes relevant contributions. Can work on her/his part and take part in the group	Generates ideas and builds upon other's ideas to develop a larger plan. Works independently to do his/her part and is invested in the other group members (e.g. helps out when needed, cares about the group product)

B.3 Pair Programming /Completing a Task in Pairs

Example rubric to assess pair programming activity: *Identifying, completing and creating patterns*.

	Learner name	#Identify Pattern	#Complete Pattern.	#Create Pattern	*Learner fulfilled role well
1.					
2.					

#Replace with suitable criteria depending on the task/problem
*Will remain the same irrespective of task/problem

Note:

Although both learners get the same mark for the first three criteria, each learner gets an individual mark for the "Learner fulfilled role well" – this is based on how well each learner contributed based on their set role.

The teacher can give most of these marks while learners are completing the activity and hence it should not require much extra time.

Each of the aspects listed in the table above, is assessed using the following key:

Aspect assessed	Beginning (1)	Developing (2)	Accomplished (3)	Exemplary (4)
Identify Pattern	• Needs assistance to identify the pattern and cannot describe the pattern in terms of the correct pattern rule(s)	• Able to identify the pattern but needs assistance to describe the pattern in terms of the correct pattern rule(s)	• Able to identify the pattern and describe the pattern in terms of the correct pattern rules with minor shortcomings	• Able to identify and fully describe the pattern in terms of the correct pattern rules
Complete Pattern	• Needs assistance to complete a pattern due to not understanding the pattern rule(s).	• Able to complete the pattern but needed help with pattern rule(s)	• Able to complete the pattern according to the rule(s) identified using 2 attempts	• Able to complete the pattern according to the pattern rule(s) identified on first attempt
Create Pattern	• Needed assistance to create the pattern and does not understand pattern rule(s)	• Create repeating patterns but needed help with pattern rule	• Able to create own pattern according to a pattern rule using 2 attempts	• Can create own pattern according to their own rule(s) on first attempt
Learner fulfilled role well	• Learner does not understand his/her role and makes no contribution or unrelated contributions	• Shares ideas or tries to fulfil her/his role, but does not work well with peer and most of the contributions are unrelated	• Tries to understand his/her role and mostly makes relevant contributions. Can work on her/his part to contribute to the solution	• Generates ideas and builds upon peer's ideas to develop a larger plan./ solve the problelm

B.4 Communication / Discussion (Digital Concepts)

Example, explaining a concept, e.g., what a robot is

An effective communicator shares information and ideas for a given purpose, task, and audience.				
Competencies	**Beginning (1)**	**Developing (2)**	**Accomplished (3)**	**Exemplary (4)**
Explaining the concept	• Learner's explanation is unclear or difficult to follow	• Learner's explanation is generally clear and easy to follow, but may be incomplete or somewhat disorganised	• Learner's explanation is well-organised and easy to follow	• Learner's explanation is well-organized, engaging, and demonstrates creativity and originality
Key information included	• Learner includes some key information but may be missing some important details.	• Learner includes key information and some details to support their explanation.	• Learner includes all key information and all relevant details to support their explanation	• Learner explains the concept in depth, demonstrating a deep understanding

B.5 Design Thinking

A process that emphasizes creativity, experimentation, and iteration to arrive at the best solution that meets user needs.				
Competencies	**Beginning (1)**	**Developing (2)**	**Accomplished (3)**	**Exemplary (4)**
Inspiration: **Learner applies creative thinking to create a product or complete a task**	• Demonstrates limited creative thinking and understanding of the problem or task	• Applies creative thinking to understand the problem or task and identifies some opportunities for innovation	• Applies creative thinking effectively to gain a deeper understanding of the problem or task and identifies significant opportunities for innovation.	• Demonstrates exceptional creative thinking and in-depth understanding of the problem or task, uncovering unique insights and opportunities for innovation
Ideation: **Learner can create own ideas to create a product or completing a task.**	• Unsure about what is expected so any idea is scattered or unfocused and ideas do not clearly connect to the problem or task.	• Generally, mimics ideas from others (rather than creating new ideas) that are related to the problem or task.	• Creates new ideas that include enough detail and that are directly related to the problem or task.	• Creates many clear ideas by considering lots of possibilities that focuses on key information and fully addresses the problem or task
Implementation: **Learner can use best ideas to create a product or complete a task.**	• Creates a product or performance, but the product has limited functionality or detail and does not clearly address the problem or the product is not useful.	• Creates a product or performance with some functionality that is somehow related to the challenge or problem.	• Uses ideas to create a product or performance with good functionality that is directly related to the problem or task.	• Creates clear ideas to create a product or performance with precision and full functionality and that fully addresses the problem or task..
Testing & Improving	• Provides minimal or no feedback and does not reflect on the quality to consider improvements or iterations	• Collects some feedback and reflects somewhat on the quality for considering minor improvements or iterations	• Collects thorough feedback, reflects accurately on the quality to inform improvements, and iterates on the solution	• Collects extensive feedback, conducts rigorous testing, and iterates on the design or solution based on feedback, leading to transformative improvements.

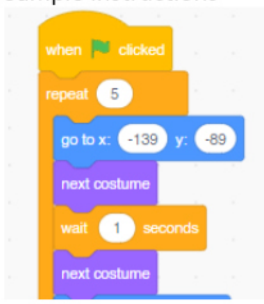
Note: All rubrics serve as examples only and may be adapted

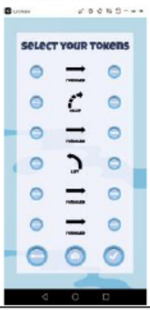
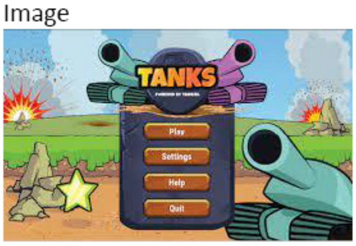
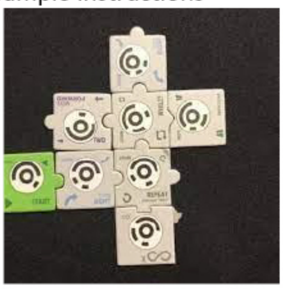
ANNEXURE C – POSSIBLE ADDITIONAL RESOURCES

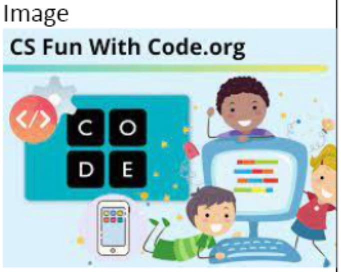
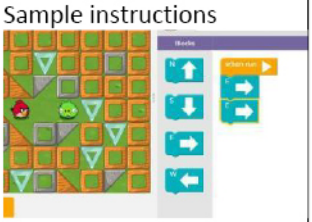
For the foundation phase, it is possible to fulfil the curriculum in its entirety unplugged (without coding software or robotics tools). However, should the school want to use coding software or educational robotics tools, they need to consider the possible impact it may have on the cognitive load (Refer to Section 2.9, Figure 2.7).

The following educational resources could be considered to support unplugged activities:

Robot Mouse		BEE Bot	
Image	Sample instructions	Image	Sample instructions
			

Scratch Junior		Scratch	
Image	Sample instructions	Image	Sample instructions
 https://www.scratchjr.org/		 https://scratch.mit.edu/	

Boats		Tanks	
Image	Sample instructions	Image	Sample instructions
		 https://www.tangiblgames.com	

Code.org		Kodable	
Image	Sample instructions	Image	Sample instructions
 https://code.org/		 https://www.kodable.com/