



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NASIONALE
SENIOR SERTIFIKAAT**

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

FEBRUARIE/MAART 2017

MEMORANDUM

PUNTE: 150

Hierdie memorandum bestaan uit 29 bladsye.

ALGEMENE INLIGTING:

- Hierdie nasienriglyne moet as die basis vir die nasiensessie gebruik word. Dit is voorberei om deur nasieners gebruik te word. Daar word na alle nasieners verwag om 'n deeglike standaardiseringsvergadering by te woon om seker te maak dat die riglyne konsekwent geïnterpreteer en tydens die nasien van die kandidate se werk toegepas word.
- Let op dat leerders wat 'n alternatiewe korrekte oplossing as wat as voorbeeld van 'n oplossing in die nasienriglyne gegee word verskaf, volle krediet vir die relevante oplossing moet kry tensy die spesifieke instruksies in die vraestel nie gevolg is nie of die vereistes van die vraag nie nagekom is nie.
- **Bylaag A, B en C** (bladsy 3–9) sluit die nasienrubriek vir elke om te gebruik vir enigeen van die twee programmeringstale in.
- **Bylaag D, E en F** (bladsy 10–17) bevat voorbeelde in programmeringskode in Java van oplossings vir VRAAG 1 tot VRAAG 3.
- **Bylaag G, H en I** (bladsy 18–29) bevat voorbeelde in programmeringskode in Delphi van oplossings vir VRAAG 1 tot VRAAG 3.
- Kopieë van **Bylaag A, B en C** (bladsy 3–9) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word.

BYLAAG A:**AFDELING A****VRAAG 1: NASIENRUBRIEK - ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
1.1	<p>Knoppie - [Vraag 1_1]</p> <p>Kry naam en van uit teksblokkie ✓ Kontroleer of spatie ingesluit is in naam-en-van string ✓ As geen spatie ingesluit ✓ Vertoon boodskap wat aandui dat spatie nie ingesluit is nie ✓</p> <p>Kry IDnommer ✓ As IDnommer-teksblokkie leeg is ✓ Vertoon boodskap wat aandui dat die teksblokkie leeg is ✓ As IDnommerstring nie uit 13 syfers bestaan nie ✓ Vertoon boodskap wat aandui dat 13 syfers nie ingesleutel is nie ✓</p> <p>As naamVan string 'n spatie insluit ✓ EN IDnommer is nie leeg nie ✓ EN IDnommer bevat 13 syfers ✓ (Beide voorwaardes getoets met AND) ✓ Maak Welkom-boodskap sigbaar ✓</p>	14	
1.2	<p>Knoppies met prentjies (ikone) wat tipe Lojaliteitskaart wys</p> <p>Knoppie Afslag: Stel globale waarde op AF ✓ Verander Afslag-opskrif na helder ('bold') ✓ en Donasie-opskrif na nie helder ('not bold') ✓</p> <p>Knoppie Donasie: Stel globale waarde op LI ✓ Verander Donasie-opskrif na helder ('bold') ✓ en Afslag-opskrif na nie helder ('not bold') ✓</p> <p>NOTA: Een punt vir 'n poging om die teks op die knoppie na bold te verander. Een punt vir 'n poging om die teks op die knoppie te herstel vanaf bold.</p>	6	

1.3	<p>Knoppie - [Vraag 1_3]</p> <p>Kry aankope se bedrag uit teksblokkie ✓ en skakel om na reële waarde ✓</p> <p>Gebruik tipe kaart se kode om te toets ('Case/Switch' of 'If):</p> <p>As 'AF' ✓</p> <p style="padding-left: 20px;">Bereken afslag as 1.5% ✓ van bedrag ✓</p> <p style="padding-left: 20px;">Stel boodskap saam ✓ wat bedrag as geldwaarde ✓ met 2 desimale plekke vertoon ✓</p> <p>As 'LI' ✓</p> <p style="padding-left: 20px;">Bereken getal $R100de \checkmark * 3 \checkmark$</p> <p style="padding-left: 20px;">Maak voorsiening vir gedeelte van R100 ✓</p> <p style="padding-left: 20px;">Stel boodskap saam ✓ wat bedrag insluit ✓</p> <p>Vertoon boodskap in teksblokkie ✓</p>	15	
1.4	<p>Knoppie - [Vraag 1_4]</p> <p>Deel 1:</p> <p>Ken die verstelwaarde P aan veranderlike toe ✓</p> <p>As Besigheidskartaat afgemerk is ✓, ken dan B aan veranderlike toe ✓</p> <p>Kry eerste 3 syfers uit IDnommer ✓ en voeg by veranderlike se inhoud ✓</p> <p>Deel 2:</p> <p>Kry laaste syfer uit die IDnommer ✓✓</p> <p>Skakel om na heelgetal ✓</p> <p>Kry karakter by heelgetal-posisie in gegewe string ✓</p> <p>Voeg karakter by syfer ✓</p> <p>Vertoon stringe wat saamgestel is in teksblokkies:</p> <p style="padding-left: 20px;">Deel 1 ✓</p> <p style="padding-left: 20px;">Deel 2 ✓</p> <p style="padding-left: 20px;">Deel (AF/LI) ✓</p>	13	
TOTAAL:		48	

BYLAAG B:**AFDELING B****VRAAG 2: NASIENRUBRIEK - OBJEK-GEÛRIENTEERDE PROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
2.1.1	Konstruktor: Definisie met drie regte parameters ✓ en regte datatipes ✓ Ken parameterwaarde toe aan kaartnommer, seLfoonnommer en lojaliteitspunte ✓ Inisialiseer getal besoeke na 0 ✓ Inisialiseer gesondVlak-attribuut na S ✓	5	
2.1.2	'Mutator' metode: setAantBesoeke-metode met parameter ✓ geen terugstuur ✓	2	
2.1.3	vermeerderLojaliteitsPunte-metode: Metode se definisie ontvang totale bedrag as parameter as reële waarde ✓ Bepaal getal lojaliteitspunte wat verdien is: Verandet nedrag na heelgetal (sonder om af te rond ✓ – volle rande) ✓ Deel deur 4 ✓ Voeg berekende punte by lojaliteitspunte-attribuut ✓	5	
2.1.4	opdateerGesondVlak-metode: Metode definisie ontvang ontvang gesondheidsbedrag en totale bedrag as parameters ✓ Bereken persentasie ✓ As persentasie minder is as 10 ✓ Ken 'S' toe aan gesondVlak-attribuut ✓ As persentasies 10 of meer en minder as 40 is ✓ Ken 'G' toe aan gesondVlak-attribuut ✓ As persentasie 40 of meer is Ken 'P' toe aan gesondVlak-attribuut ✓	7	

2.1.5	<p>isReg-metode: (Metode se definisie was voorsien)</p> <p>Lus deur die syfers van die IDnommer ✓ Verwyder die 0-syfers ✓ uit die string Inisialiseer veranderlike vir som ✓ As ewe getal syfers oor is in string ✓ Lus regte aantal keer ✓ Tel tweesyferwaardes by som-veranderlike van linkerkant van string af ✓ As onewe getal syfers oor is in string ✓ Voeg eerste syfer van linkerkant af in string as eensyferwaarde by som ✓ Lus regte aantal keer ✓ Tel tweesyferwaardes by som-veranderlike ✓ As som gelyk is aan toegangskode-parameter ✓ stuur 'true' terug ✓ anders stuur false terug ✓</p>	13	
2.1.6	<p>identifieerSterKoper-metode:</p> <p>Lee stringveranderlike – as nie Ster-koper nie ✓ As (lojaliteitspunte > 2000 ✓ EN getal besoeke > 10) ✓ OF ✓ (gesondVlak is 'P') ✓ Stel stringveranderlike op 'STER-koper' ✓ Stuur stringveranderlike terug</p>	6	

VRAAG 2: NASIENRUBRIEK – vervolg

2.2.1	<p>Knoppie – [2.2.1 – Kontroleer toegangskode] Kry kaartnommer uit kombinasieblokkie ✓ Kry selfoonnommer uit byskrif ('label') ✓ Kry toegangskode uit teksblokkie ✓ Kry lojaliteitspunte uit byskrif en skakel om na heelgetal ✓ Instansieer objek ✓ deur regte argumente in regte volgorde te stuur ✓ (6)</p> <p>As toegangskode reg is Toets of toegangskode reg is ✓ deur van isReg-metode gebruik te maak ✓ Vertoon boodskap dat toegangskode reg is</p> <p>Lees tekslêer en verwerk data {Delphi: AssignFile, Reset en CloseFile Java: Skep objek uit uit leer te lees} ✓ Inisialiseer veranderlikes vir tellers en som ✓ Lus deur lêer ✓ Lees kaartnommer ✓ Lees totale bedrag wat spandeer is en skakel om na 'double' datatipe ✓ Lees gesondheidsbedrag en skakel om na 'double' datatipe ✓ Toets of regte kaartnommer ✓ Tel totale bedrag by somTotaal ✓ Tel gesondheidsbedrag by somGesond ✓ Inkrementeer Teller ✓ Roep setAantBesoeke-metode ✓ Roep vermeerderLojaliteitsPunte-metode – stuur sumTotaal as argument ✓ Roep opdateerGesondVlak-metode – stuur somTotaal en somgesond as argumente ✓</p> <p>eindig lus Maak knoppie vir V2.2.2 aktief ✓</p> <p>As toegangskode NIE reg is NIE Vertoon boodskap dat toegangskode nie reg is nie ✓ Maak die teksblokkie vir toegangskode skoon ✓ (18)</p>	24	
2.2.2	<p>Knoppii – [2.2.2 – Vertoon kaarthouer se besonderhede] Vertoon objek se besonderhede in die afvoerarea ✓ deur van toString ✓ gebruik te maak Vertoon koper se status deur van die identifiseerSterKoper-metode gebruik te maak ✓</p>	3	
TOTAAL:		65	

BYLAAG C:**AFDELING C****VRAAG 3: NASIENRUBRIEK - PROBLEEMOPLOSSING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
3.1	Knoppie [3.1 – Vertoon uitleg] Buitenste lus vir rye ✓ Binneste lus vir kolomme ✓ Vertoon simbool uit tweedimensionele skikking ✓ In ry ✓ en kolom ✓	5	
3.2	Kombinasieblokkie - Restaurante Kry winkel as karakter uit kombinasieblokkie ✓ Inisialiseer veranderlikes vir posisie (ry en kolom ✓) Lus deur die rye ✓ As die winkel een van karakters A-E is (links) ✓ Stel kolom vir winkel op 0 (eerste kolom) ✓ Stel kolom van vriend op 1 (tweede kolom) ✓ As die winkel een van karakters F-J is (regs) ✓ Stel winkel kolom op laaste kolom van arrSentrum ✓ Stel vriend kolom op een minder as winkel kolom ✓ Toets of karakter by huidige posisie in skikking ✓ gelyk is aan die karakter wat geselekteer is ✓ Vervang X met # ✓ Dateer vertoon op ✓ Eindig lus	13	
3.3	Knoppie [3.3 – Spoor vriend(e) en naaste vriende op] Inisialiseer naaste afstand na groot waarde ✓ Buitenste lus vir rye ✓ Binneste lus vir kolomme/as regterkant ✓ As karakter by posisie # is ✓ Bepaal verskil in ry-posisies ✓ -positiewe getal ✓ Bepaal verskil in kolom-posisies ✓ -positiewe getal ✓ Bereken nuweAfstand ✓ As vriend in A-E is ✓ vertoon winkelnaam ✓ As vriend in F-J is ✓ vertoon winkelnaam ✓ If nuweAfstand <= naaste afstand ✓ Voeg winkelnaam by afvoerboodskap ✓ Vervang naaste met nuweAfstand ✓ Vertoon winkelnaam ✓ en afstand * 4.5 ✓ Vertoon afvoerboodskap wat naaste vriende aandui ✓	19	
TOTAAL:		37	

OPSOMMING VAN LEERDERPUNTE:

SENTRUMNOMMER:		EKSAMENNOMMER:		
	AFDELING A	AFDELING B	AFDELING C	
	VRAAG 1	VRAAG 2	VRAAG 3	GROOTTOTAAL
MAKS.PUNTE	48	65	37	150
LEERDERPUNTE				

BYLAAG D: OPLOSSING VIR VRAAG 1: JAVA

```
// Kode wat voorsien is
    String tipeKaart = "";
    String karakString = "!@#$%]&*^~";

//*****
// Vraag 1.1
// *****
private void btnVraag1_1ActionPerformed(java.awt.event.ActionEvent evt)
{
    String naamVan = txfNaamVan.getText();
    String idNommer = txfIDNommer.getText();
    boolean geenSpasie = naamVan.contains(" ");
    boolean geenId = idNommer.length() == 0;
    boolean idLen = idNommer.length() != 13;
    if (!geenSpasie) {
        JOptionPane.showMessageDialog(rootPane, "Die naam-en-van-veld bevat
            nie 'n spasie nie.");
    }
    if (geenId) {
        JOptionPane.showMessageDialog(rootPane, "'n ID-nommer moet
            ingesleutel word.");
    } else if (idLen) {
        JOptionPane.showMessageDialog(rootPane, "Die ID-nommer bestaan nie
            uit 13 syfers nie.");
    }
    if (geenSpasie && !idLen && !geenId) {
        lblWelkomBoodskap.setVisible(true);
    }
}
// *****
// Vraag 1.2
// *****
private void btn1_2AfslagActionPerformed(java.awt.event.ActionEvent evt)
{
    tipeKaart = "AF";
    lblAfslag.setFont(new java.awt.Font("Tahoma", 1, 12));
    lblDonasie.setFont(new java.awt.Font("Tahoma", 0, 12));
}

private void btn1_2DonasieActionPerformed(java.awt.event.ActionEvent
evt)
{
    tipeKaart = "LI";
    lblDonasie.setFont(new java.awt.Font("Tahoma", 1, 12));
    lblAfslag.setFont(new java.awt.Font("Tahoma", 0, 12));
}
}
```

```
// *****
// Vraag 1.3
// *****
private void btnVraag1_3ActionPerformed(java.awt.event.ActionEvent evt) {
    double bedrag = Double.parseDouble(txfBedrag.getText());
    String boodskap = "";
    switch (tipeKaart) {
        case "AF":
            double afslag = bedrag * 0.015;
            boodskap = "1.5% afslag op jou aankoopbedrag is " +
                String.format("R %.2f", afslag) + ".";
            break;
        case "LI":
            double bydrae = Math.ceil(bedrag / 100.0) * 3;
            boodskap = "\n Bedrag van " + String.format("R %.2f",
                bydrae) + " sal aan liefdadigheid geskenk word.";
            break;
    }
    txfBoodskap.setText(boodskap);
}

// *****
// Vraag 1.4
// *****
private void btnVraag1_4ActionPerformed(java.awt.event.ActionEvent evt)
{
    String bKaart = "P";
    if (chkBesigheid.isSelected()) {
        bKaart = "B";
    }
    String deel1 = bKaart + txfIDNommer.getText().substring(0, 3);
    char syfer = txfIDNommer.getText().charAt(12);
    int pos = Integer.parseInt(syfer + "");
    String deel2 = syfer + "" + karakString.charAt(pos) + "";

    txfDeel1.setText(deel1 + "");
    txfDeel2.setText(deel2 + "");
    txfDeel3.setText(tipeKaart);
}
```

BYLAAG E: OPLOSSING VIR VRAAG 2: JAVA**OPLOSSING VIR VRAAG 2: OBJEKKLAS**

```
package Vraag2Package;

public class KaartHouer {

    //Kode wat voorsien is
    private String kaartNommer;
    private String selNommer;
    private int aantbesoeke;
    private int lojaliteitsPunte;
    private char gesondVlak;

    //*****
    // Vraag 2.1.1
    //*****
    public KaartHouer(String kaartNommer, String selNommer,
        int lojaliteitsPunte) {
        this.kaartNommer = kaartNommer;
        this.selNommer = selNommer;
        this.lojaliteitsPunte = lojaliteitsPunte;
        this.aantbesoeke = 0;
        this.gesondVlak = 'S';
    }
    //*****
    // Vraag 2.1.2
    //*****
    public void setAantBesoeke(int aantbesoeke) {
        this.aantbesoeke = aantbesoeke;
    }
    //*****
    // Vraag 2.1.3
    //*****
    public void vermeerderLojaliteitsPunte(double totaal) {
        lojaliteitsPunte = lojaliteitsPunte + (int) totaal / 4;
    }
    //*****
    // Vraag 2.1.4
    //*****
    public void opdateerGesondVlak(double totaalGesond, double totaalBedrag)
    {
        double persent = totaalGesond / totaalBedrag * 100;
        gesondVlak = 'S';

        if (persent >= 10 && persent < 40) {
            gesondVlak = 'G';
        }
        if (persent >= 40) {
            gesondVlak = 'P';
        }
    }
}
```

```

//*****
// Vraag 2.1.5
//*****
public boolean isReg(int toegangNom) { //Gegee
    String cNommer = "";
    for (int tel = 0; tel < selNommer.length(); tel++) {
        if (selNommer.charAt(tel) != '0')
        {
            cNommer = cNommer + selNommer.charAt(tel);
        }
    }
    int som = 0;
    if (cNommer.length() % 2 == 0) {
        for (int tel = 0; tel < cNommer.length(); tel = tel + 2) {
            som = som + Integer.parseInt(cNommer.substring(tel, tel+2));
        }
    } else {
        som = Integer.parseInt(cNommer.substring(0, 1));
        for (int tel = 1; tel < cNommer.length(); tel = tel + 2) {
            som = som + Integer.parseInt(cNommer.substring(tel, tel +2));
        }
    }
    if (som == toegangNom) {
        return true;
    } else {
        return false;
    }
}

//*****
// Vraag 2.1.6
//*****
public String identifiseerSterKoper() {
    String ster = "";
    if ((lojaliteitsPunte > 2000 && aantbesoeke > 10) || (gesondVlak ==
        'P')) {
        ster = "STER-koper";
    }
    return ster;
}

//*****
// Kode wat voorsien is
//*****
//
public String toString() {
    return kaartNommer + "\nKontaknommer: " + selNommer +
        "\n\nOpgedateerde getal lojaliteitspunte: " + lojaliteitsPunte +
        "\nGetal besoeke: " + aantbesoeke +
        "\nGesondheidsevalueringstatus: " + gezondVlak;
}
}

```

GGK-KLAS: VRAAG2_OPLOSSING

```

//*****
// Kode wat voorsien is
//*****

    KaartHouer objKaartHouer = null;

    public Vraag2_Memo() {
        initComponents();
        setLocationRelativeTo(this);
        btnVraag222.setEnabled(false);
    }

//*****
// Vraag 2.2.1
//*****
private void btnVraag221ActionPerformed(java.awt.event.ActionEvent evt)
{
    String kaartNom = "" + cmbKaartNommers.getSelectedItem();
    String selNom = lblSelNommer.getText();
    String toegangNommer = txfKode.getText();
    int toegangNom = Integer.parseInt(txfKode.getText());
    int huidigePunte = Integer.parseInt(lblLojaliteitsPunte.getText());
    objKaartHouer = new KaartHouer(kaartNom, selNom, huidigePunte);
    if (objKaartHouer.isReg(toegangNom)) {
        JOptionPane.showMessageDialog(rootPane, "Die toegangskode is reg.");
        try {
            Scanner scLeer = new Scanner(new
                FileReader("DataJanuarie2017.txt"));
            double totaalBedrag = 0;
            double totaalGesond = 0;
            int teller = 0;
            while (scLeer.hasNext()) {
                String kaart = scLeer.nextLine();
                double bedrag = Double.parseDouble(scLeer.nextLine());
                double gesondBedrag = Double.parseDouble(scLeer.nextLine());
                if (kaart.equalsIgnoreCase(kaartNom)) {
                    totaalBedrag = totaalBedrag + bedrag;
                    totaalGesond = totaalGesond + gesondBedrag;
                    teller++;
                } // if
            } // while
            objKaartHouer.setAantBesoeke(teller);
            objKaartHouer.vermeerderLojaliteitsPunte(totaalBedrag);
            objKaartHouer.opdateerGesondVlak(totaalGesond, totaalBedrag);
        } catch (FileNotFoundException e) {
            JOptionPane.showMessageDialog(rootPane, e);
        } // try
        btnVraag222.setEnabled(true);
    } else {
        JOptionPane.showMessageDialog(rootPane, "Verkeerde
            toegangskode.");
    }
    txfKode.setText("");
}
}

```

```
//*****  
// Vraag 2.2.2  
//*****  
private void btnVraag222ActionPerformed(java.awt.event.ActionEvent evt)  
{  
    txaAfvoer.setText(objKaartHouer.toString());  
    txaAfvoer.append("\n" + objKaartHouer.identifiseerSterKoper());  
}  
//*****  
// Kode wat voorsien is  
//*****  
public void opdateerVanPunte() {  
    if (cmbKaartNommers.getSelectedIndex() == 0) {  
        lblLojaliteitsPunte.setText("2130");  
        lblSelNommer.setText("0812345678");  
    }  
    if (cmbKaartNommers.getSelectedIndex() == 1) {  
        lblLojaliteitsPunte.setText("5723");  
        lblSelNommer.setText("0822001100");  
    }  
    if (cmbKaartNommers.getSelectedIndex() == 2) {  
        lblLojaliteitsPunte.setText("12908");  
        lblSelNommer.setText("0740998877");  
    }  
    if (cmbKaartNommers.getSelectedIndex() == 3) {  
        lblLojaliteitsPunte.setText("500");  
        lblSelNommer.setText("0720951083");  
    }  
    txaAfvoer.setText("");  
    txfkode.setText("");  
    btnVraag222.setEnabled(false);  
}
```

BYLAAG F: OPLOSSING VIR VRAAG 3: JAVA

```
//*****
// Kode wat voorsien is
//*****
char[][] arrSentrum = {{'A', 'X', 'O', '*', 'O', 'X', 'J'}, {'B', 'X',
'O', 'O', 'O', 'X', 'I'}, {'C', 'X', 'O', 'O', 'O', 'X', 'H'}, {'D',
'X', 'O', 'O', 'O', 'X', 'G'}, {'E', 'X', 'O', 'O', 'O', 'X', 'F'}};
//*****
// Vraag 3.1
//*****
private void btnVertoonActionPerformed(java.awt.event.ActionEvent evt)
{
    txaVraag31.setText("\n");
    for (int ry = 0; ry < arrSentrum.length; ry++) {
        for (int kolom = 0; kolom < arrSentrum[0].length; kolom++) {
            txaVraag31.append(String.format("%5s",
                arrSentrum[ry][kolom]));
        }
        txaVraag31.append("\n");
    }
}

//*****
// Vraag 3.2
//*****
private void cmbWinkelsActionPerformed(java.awt.event.ActionEvent evt) {
    char winkel = ((String) cmbWinkels.getSelectedItem()).charAt(0);
    int kolWinkel = 0;
    int kolVriend = 0;
    for (int ry = 0; ry < arrSentrum.length; ry++) {
        if (winkel >= 'A' && winkel <= 'E') {
            kolWinkel = 0;
            kolVriend = 1;
        } else {
            kolWinkel = arrSentrum[0].length - 1;
            kolVriend = kolWinkel - 1;
        }
        if (arrSentrum[ry][kolWinkel] == winkel) {
            arrSentrum[ry][kolVriend] = '#';
        }
    }
    btnVertoon.doClick();
}
```



```
//*****  
// Vraag 3.3  
//*****  
  
private void btnOpspoorActionPerformed(java.awt.event.ActionEvent evt) {  
    int afstand = 0;  
    int nuweAfstand = 100;  
    String boodskap = "Jou naaste vriend(e) is by: ";  
    String boodsk = "";  
    char winkel = ' ';  
    txaVraag33.setText("");  
    for (int ry = 0; ry < arrSentrum.length; ry++) {  
        for (int kol = 0; kol < arrSentrum[0].length; kol++) {  
            if (arrSentrum[ry][kol] == '#') {  
                int verskilRy = Math.abs(ry - rJouRy);  
                int verskilKol = Math.abs(kol - cJouKolom);  
                afstand = verskilRy + verskilKol + 1;  
                if (kol == 5) {  
                    winkel = arrSentrum[ry][kol + 1];  
                } else { //1 rest name  
                    winkel = arrSentrum[ry][kol - 1];  
                }  
                txaVraag33.append("Restaurant " + winkel + " - naastenby " +  
                    (afstand * 4.5) + " meter\n");  
                if (afstand <= nuweAfstand) {  
                    nuweAfstand = afstand;  
                    boodsk = boodsk + " " + winkel;  
                }  
            }  
        }  
    }  
    txaVraag33.append("\n" + boodskap + boodsk);  
}
```

BYLAAG G: OPLOSSING VIR VRAAG 1: DELPHI

```
unit Vraag1_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, ExtCtrls, pngimage, Math;

type
  TfrmVraag1 = class(TForm)
    pnlOpskrif: TPanel;
    gpbVraag11: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    edtNaam: TEdit;
    edtIDNommer: TEdit;
    Label3: TLabel;
    btnVraag11: TButton;
    gpbVraag12: TGroupBox;
    Label4: TLabel;
    lblAfslag: TLabel;
    lblDonasie: TLabel;
    imgAfslag: TImage;
    imgDonasie: TImage;
    gpbVraag13: TGroupBox;
    Label7: TLabel;
    edtBedrag: TEdit;
    edtVertoon: TEdit;
    btnVraag13: TButton;
    gpbVraag14: TGroupBox;
    ckbBesigheid: TCheckBox;
    Label9: TLabel;
    edtDeel1: TEdit;
    Label10: TLabel;
    edtDeel2: TEdit;
    Label11: TLabel;
    edtDeel3: TEdit;
    btnVraag14: TButton;
    lblWelkomboodskap: TLabel;
    Label5: TLabel;
    procedure btnVraag11Click(Sender: TObject);
    procedure imgAfslagClick(Sender: TObject);
    procedure imgDonasieClick(Sender: TObject);
    procedure btnVraag13Click(Sender: TObject);
    procedure btnVraag14Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
```

```

var
  sNaam, sIDNommer: string;

//*****
// Kode wat voorsien is
//*****
  frmVraag1 : TfrmVraag1;
  tipeKaart : string; // voorsien
  karakString: string = '!@#%]&*^~'; // voorsien

implementation

{$R *.dfm}

//*****
// Vraag 1.1
//*****
procedure TfrmVraag1.btnVraag1Click(Sender: TObject);
var
  bGeenSpasie, bGeenID, bIDLen: boolean;
begin
  // Vraag 1.1
  sNaam := edtNaam.Text;
  bGeenSpasie := pos(' ', sNaam) = 0;
  if bGeenSpasie then
    showMessage('Die naam-en-van-veld bevat nie 'n spasie nie.');
```

```

  sIDNommer := edtIDNommer.Text;
  bGeenID := length(sIDNommer) = 0;
  bIDLen := length(sIDNommer) <> 13;
  if bGeenID then
    showMessage(''n ID-nommer moet ingesleutel word.')
```

```

  else
    if bIDLen then
      showMessage('Die ID-nommer bestaan nie uit 13 syfers nie.');
```

```

  if (bGeenSpasie = false) AND (bGeenID = false) AND (bIDLen = false)
then
  lblWelkomBoodskap.visible := true;
end;

//*****
// Vraag 1.2
//*****
procedure TfrmVraag1.imgDonasieClick(Sender: TObject);
begin
  // Vraag 1.2
  lblDonasie.Font.Style := [fsBold];
  lblAfslag.Font.Style := [];
  tipeKaart := 'LI';
end;

procedure TfrmVraag1.imgAfslagClick(Sender: TObject);
begin
  // Vraag 1.2
  lblAfslag.Font.Style := [fsBold];
  lblDonasie.Font.Style := [];
  tipeKaart := 'AF'
end;

```

```

//*****
// Vraag 1.3
//*****
procedure TfrmVraag1.btnVraag13Click(Sender: TObject);
var
  rAfslag, rDonasie: real;
begin
  // Vraag 1.3
  if tipeKaart = 'AF' then
    begin
      rAfslag := 1.5 / 100 * StrToFloat(edtBedrag.Text);
      edtVertoon.Text := '1.5% afslag op jou aankoopbedrag is ' +
        FloatToStrF(rAfslag, ffCurrency, 6, 2) + '.';
    end
  else
    begin
      rDonasie := Math.Ceil(StrToFloat(edtBedrag.Text) / 100) * 3;
      edtVertoon.Text := ''n Bedrag van ' + FloatToStrF(rDonasie,
        ffCurrency, 6,2) + ' sal aan liefdadigheid geskenk word.';
    end
  end;
//*****
// Vraag 1.4
//*****
procedure TfrmVraag1.btnVraag14Click(Sender: TObject);
var
  sSimbool: string;
  iPosisie: integer;
  bKaart : char;
begin
  // Vraag 1.4
  if ckbBesigheid.Checked then
    bKaart := 'B'
  else
    bKaart := 'P';
  iPosisie := StrToInt(sIDNommer[length(sIDNommer)]);
  sSimbool := karakString[iPosisie + 1];
  edtDeel1.Text := bKaart + copy(sIDNommer, 1, 3);
  edtDeel2.Text := IntToStr(iPosisie) + sSimbool;
  edtDeel3.Text := tipeKaart;
end;

//*****
// Kode wat voorsien is
//*****
procedure TfrmVraag1.FormActivate(Sender: TObject);
begin
  CurrencyString := 'R';
end;

end.

```

BYLAAG H: OPLOSSING VIR VRAAG 2: DELPHI**OBJEKKLAS:**

```
unit KaartHouer_U;

interface
  uses SysUtils, Math;
type
  TKaartHouer = class(TObject)
  private
    fKaartNommer: string;
    fSelNommer: string;
    fAantBesoeke: integer;
    fLojaliteitsPunte: integer;
    fGesondVlak: char;
  public
    constructor create(sKaartNommer, sSelNommer: string; iLojaliteitsPunte:
integer);
    procedure setAantbesoeke(iAantBesoeke: integer);
    Procedure vermeerderLojaliteitsPunte(rTotaal: real);
    procedure opdateerGesondVlak(rGesond, rTotaal: real);
    function isReg(sToegangsKode:string): boolean;
    function identifiseerSterKoper: string;
    function toString: string;
  end;

implementation

{ TKaartHouer }

// *****
// Vraag 2.1.1
// *****
constructor TKaartHouer.create(sKaartNommer, sSelNommer: string;
    iLojaliteitsPunte: integer);
begin
  fKaartNommer := sKaartNommer;
  fSelNommer := sSelNommer;
  fLojaliteitsPunte := iLojaliteitsPunte;
  fAantBesoeke := 0;
  fGesondVlak := 'S';
end;

// *****
// Question 2.1.2
// *****
procedure TKaartHouer.setAantbesoeke(iAantBesoeke: integer);
begin
  fAantBesoeke := iAantBesoeke;
end;
```

```
// *****
// Question 2.1.3
// *****
procedure TKaartHouer.vermeerderLojaliteitsPunte(rTotaal: real);
var
  iLojaliteitsPunte: integer;
begin
  fLojaliteitsPunte := fLojaliteitsPunte + trunc(rTotaal) div 4;
end;

// *****
// Question 2.1.4
// *****
procedure TKaartHouer.opdateerGesondVlak(rGesond, rTotaal: real);
var
  rPersent: real;
begin
  rPersent := rGesond/rTotaal * 100;
  if rPersent < 10 then
    fGesondVlak:='S'
  else
    if rPersent >= 40 then
      fGesondVlak := 'P'
    else
      fGesondVlak := 'G'
  end;
end;

// *****
// Question 2.1.5
// *****
// Definisie is voorsien

function TKaartHouer.isReg(sToegangsKode: string): boolean;
var
  a: integer;
  b,iSom: integer;
  sSelNommer : string;
begin
  iSom := 0;
  sSelNommer := fSelNommer;
  for a := length(sSelNommer) downto 1 do
    if sSelNommer[a] = '0' then
      delete(sSelNommer,a,1);
  if length(sSelNommer) mod 2 = 0 then
    for b := 1 to (length(sSelNommer) div 2) do
      iSom := iSom + StrToInt(copy(sSelNommer,2*b-1,2));
  if length(sSelNommer) mod 2 <> 0 then
    begin
      iSom := StrToInt(sSelNommer[1]);
      for b := 1 to (length(sSelNommer) div 2) do
        iSom := iSom + StrToInt(copy(sSelNommer,2*b,2));
    end;
  if StrToInt(sToegangsKode) = iSom then
    result := true
  else
    result := false;
end;
```

```
// *****  
// Question 2.1.6  
// *****  
function TKaartHouer.identifiseerSterKoper: string;  
begin  
    result := '';  
    if ((fLojaliteitsPunte > 2000) AND ( fAantBesoeke > 10)) OR (  
fGesondVlak = 'P') then  
        result := 'STER-koper'  
end;  
  
// *****  
// Kode wat voorsien is  
// *****  
function TKaartHouer.toString: string;  
begin  
    result:= fKaartNommer + #13 + 'Kontaknommer: ' + fSelNommer + #13 + #13  
        + 'Opgedateerde getal lojaliteitspunte: ' +  
        IntToStr(fLojaliteitsPunte)+#13+'Getal besoeke: '  
        +IntToStr(fAantBesoeke)+#13+'Gesondheidsevalueringstatus: ' +  
        fGesondVlak;  
end;  
  
end.
```

HOOFVORMEENHEID: VRAAG2_U.PAS

```
unit Vraag2_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, ExtCtrls, ComCtrls, KaartHouer_U;

type
  TfrmVraag2 = class(TForm)
    Panel1: TPanel;
    gbpVraag2_1: TGroupBox;
    gpbVraag2_2: TGroupBox;
    Label1: TLabel;
    lblKaartNommer: TLabel;
    cmbKaartnommers: TComboBox;
    Panel2: TPanel;
    Label2: TLabel;
    lblLojaliteitsPunte: TLabel;
    lblSelNommer: TLabel;
    Panel3: TPanel;
    Label3: TLabel;
    edtKode: TEdit;
    btnVraag221: TButton;
    redAfvoer: TRichEdit;
    btnVraag222: TButton;
    procedure cmbKaartnommersChange(Sender: TObject);
    procedure btnVraag221Click(Sender: TObject);
    procedure btnVraag222Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag2: TfrmVraag2;
  objKaartHouer: TKaartHouer; // Voorsien
  myLeer: Textfile;

implementation

{$R *.dfm}

// *****
// Vraag 2.2.1
// *****

procedure TfrmVraag2.btnVraag221Click(Sender: TObject);
var
  sKaartHouer: string;
  sSelNommer: string;
  sLojaliteitsPunte: string;
  sToegang: string;
```



```

sReelEen, sReelTwee, sReelDrie: string;

rTotaalSpandeer, rTotaalGesond: real;
iBesoeke: integer;
begin
sKaartHouer := cmbKaartNommers.items[cmbKaartNommers.ItemIndex];
sSelNommer := lblSelNommer.Caption;
sLojaliteitsPunte := lblLojaliteitsPunte.Caption;

objKaartHouer := TKaartHouer.create(sKaartHouer, sSelNommer,
  StrToInt(sLojaliteitsPunte));
sToegang := edtKode.Text;
if objKaartHouer.isReg(sToegang) then
begin
  showMessage('Die toegangskode is reg. ');
  rTotaalSpandeer := 0;
  rTotaalGesond := 0;
  iBesoeke := 0;
  assignFile(myLeer, 'DataJanuarie2017.txt');
  reset(myLeer);
  while not eof(myLeer) do
  begin
    readln(myLeer, sReelEen);
    readln(myLeer, sReelTwee);
    readln(myLeer, sReelDrie);
    if sReelEen = sKaartHouer then
    begin
      rTotaalSpandeer := rTotaalSpandeer + StrToFloat(sReelTwee);
      rTotaalGesond := rTotaalGesond + StrToFloat(sReelDrie);
      inc(iBesoeke);
    end; // end if
  end; // end while
  closeFile(myLeer);

  objKaartHouer.vermeerderLojaliteitsPunte(rTotaalSpandeer);
  objKaartHouer.setAantBesoeke(iBesoeke);
  objKaartHouer.opdateerGesondVlak(rTotaalGesond, rTotaalSpandeer);
  btnVraag222.Enabled := true;
end // end isReg
else
begin
  showMessage('Verkeerde toegangskode. ');
  edtKode.Clear;
end;
end;

// *****
// Vraag 2.2.2
// *****

procedure TfrmVraag2.btnVraag222Click(Sender: TObject);
begin
  redAfvoer.Clear;
  redAfvoer.Lines.Add(objKaartHouer.toString);
  redAfvoer.Lines.Add(objKaartHouer.identifiseerSterKoper);
end;

```

```
// *****  
// Kode wat voorsien is  
// *****  
  
procedure TfrmVraag2.cmbKaartnommersChange(Sender: TObject);  
begin  
  edtKode.Clear;  
  redAfvoer.Clear;  
  btnVraag222.Enabled := false;  
  case cmbKaartNommers.ItemIndex of  
    0:  
      begin  
        lblLojaliteitsPunte.Caption := '2130';  
        lblSelNommer.Caption := '0812345678';  
      end;  
  
    1:  
      begin  
        lblLojaliteitsPunte.Caption := '5723';  
        lblSelNommer.Caption := '0822001100';  
      end;  
  
    2:  
      begin  
        lblLojaliteitsPunte.Caption := '12908';  
        lblSelNommer.Caption := '0740998877';  
      end;  
  
    3:  
      begin  
        lblLojaliteitsPunte.Caption := '500';  
        lblSelNommer.Caption := '0720951083';  
      end;  
  end;  
  
end;  
  
end.
```

BYLAAG I: OPLOSSING VIR VRAAG 3: DELPHI

```
unit Vraag3_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, ComCtrls, ExtCtrls, Grids;

type
  TfrmVraag3 = class(TForm)
    GroupBox1: TGroupBox;
    Panell: TPanel;
    btnVraag31: TButton;
    btnVraag33: TButton;
    cmbPlekke: TComboBox;
    GroupBox2: TGroupBox;
    redAfvoer: TRichEdit;
    Labell: TLabel;
    redGrid: TRichEdit;
    procedure btnVraag31Click(Sender: TObject);
    procedure cmbPlekkeChange(Sender: TObject);
    procedure btnVraag33Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag3: TfrmVraag3;

//*****
// Kode wat voorsien is
//*****

  arrSentrum:array[1..5,1..7] of string
= (('A', 'X', 'O', '*', 'O', 'X', 'J'), ('B', 'X', 'O', 'O', 'O', 'X', 'I'), ('C', 'X', 'O', 'O', 'O', 'X', 'H'), ('D', 'X', 'O', 'O', 'O', 'X', 'G'), ('E', 'X', 'O', 'O', 'O', 'X', 'F'));

  arrPlek:array[1..10] of string;
implementation
{$R *.dfm}
```

```
// *****
// Vraag 3.1
// *****
procedure TfrmVraag3.btnVraag31Click(Sender: TObject);
var
  iRy, iKolom: integer;
  sRy      : string;
begin
  redGrid.Lines.clear;
  redGrid.Lines.add(' ');

  for iRy := 1 to 5 do
  begin
    sRy := '  ';
    for iKolom := 1 to 7 do
      sRy := sRy + arrSentrum[iRy,iKolom] + #9;
    redGrid.Lines.add(sRy);
    redGrid.Lines.add(' ');
  end;
end;

// *****
// Vraag 3.2
// *****
procedure TfrmVraag3.cmbPlekkeChange(Sender: TObject);
var
  iRyChange:integer;
begin
  if cmbPlekke.Items[cmbPlekke.ItemIndex][1] in ['A'..'E'] then
  begin
    iRyChange:=cmbPlekke.ItemIndex + 1;
    arrSentrum[iRychange,2] := '#';
  end
  else
  begin
    iRyChange:=10-cmbPlekke.ItemIndex ;
    arrSentrum[iRychange,6] :='#';
  end;
  btnVraag31.Click;
end;
```

```
// *****
// Vraag 3.3
// *****
procedure TfrmVraag3.btnVraag33Click(Sender: TObject);
var
  iRy,a : integer;
  iKolom,iTeller: integer;
  rAfstand,rKleinste: real;
  sPlek:string;
begin
  redAfvoer.Clear;
  rKleinste := 100;
  iTeller := 1;
  sPlek := '';
  for iRy := 1 to 5 do
    begin
      if arrSentrum[iRy,2] = '#' then
        begin
          rAfstand := 4.5*(iRy+1)+4.5;
          if rAfstand <= rKleinste then
            begin
              inc(iTeller);
              rKleinste:=rAfstand;
              arrPlek[iTeller]:=arrSentrum[iRy,1];
            end;
          redAfvoer.Lines.Add('Restaurant '+ arrSentrum[iRy,1]+' -
            ongeveer '+FloatToStrF(rAfstand,ffFixed,4,1)+ ' meter');
        end;
      if arrSentrum[iRy,6] = '#' then
        begin
          rAfstand := 4.5*(iRy+1)+4.5;
          if rAfstand <= rKleinste then
            begin
              inc(iTeller);
              rKleinste := rAfstand;
              arrPlek[iTeller]:=arrSentrum[iRy,7];
            end;
          redAfvoer.Lines.Add('Restaurant '+ arrSentrum[iRy,7]+' -
            ongeveer '+FloatToStrF(rAfstand,ffFixed,4,1)+ ' meter');
        end;
    end;
  for a := 1 to iTeller do
    sPlek := sPlek + ' ' +arrPlek[a];

  redAfvoer.Lines.Add(' ');
  redAfvoer.Lines.Add('Jou naaste vriend(e) is by: '+ sPlek);
end;

end.
```