basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

NATIONAL SENIOR CERTIFICATE

GRADE 12

INFORMATION TECHNOLOGY P1

NOVEMBER 2014 (2)

MEMORANDUM

MARKS: 150

This memorandum consists of 28 pages.

GENERAL INFORMATION:

- These marking guidelines are to be used as the basis for the marking session. They were prepared for use by markers, all of whom are required to attend a rigorous standardisation meeting to ensure that the guidelines are consistently interpreted and applied in the marking of candidates' scripts.

- Note that candidates who provide an alternate correct solution to that given in the marking guidelines will be given full credit for the relevant question, unless the specific instructions in the paper was not followed or the requirements of the question was not met.

- **Annexures A, B** and **C** (pages 3–7) include the marking grid for each question for using either one of the two programming languages.

- **Annexures D, E** and **F** (pages 8–16) contain examples of solutions for Java for Questions 1 to 3 in programming code.

- **Annexures G, H** and **I** (pages 17–28) contain examples of solutions for Delphi for Questions 1 to 3 in programming code.

- Copies of **Annexures A, B** and **C** (pages 3–7) should be made for each candidate and completed during the marking session.

ANNEXURE A:

SECTION A:

QUESTION 1: MARKING GRID – GENERAL PROGRAMMING SKILLS

CENTRE NUMBER:	**EXAMINATION NUMBER:**		
QUESTION	**DESCRIPTION**	**MAX. MARKS**	**LEARNER'S MARKS**
1.1	If check box is selected change the font colour of the checkbox to green ✓ Change the caption/text on the button ✓ Else ✓ change the font colour of the checkbox to red ✓	**4**	
1.2	Extract user name from text box ✓ Extract password from text box ✓ Initialise counter variables or Boolean variables ✓ Loop through the password ✓ Check if character at each position is a letter ✓ If letter change counter/Boolean variable✓ Check if character at each position is a digit ✓ If letter change counter/Boolean variable ✓ Check if at least one letter ✓, one number ✓, no spaces ✓ and the length >= 8 ✓ Disable the button ✓ Set the picture label to be visible✓ Else ✓ Display message ✓ Clear the password text box ✓	**17**	
1.3.1	Extract the system date ✓ Compile the date ✓using the correct format ✓ Extract the e-mail address from the combo box ✓ Extract the message from the output area ✓ Display the date in the output area ✓ Display the e-mail address in the output area ✓ Display the open line and the message in the output area ✓	**8**	
1.3.2	Open the existing text file ✓✓ Write string to the text file ✓ in the correct format ✓ Display a message in a dialog box ✓ Close the file ✓ Error handling used for writing to a text file ✓	**7**	
1.4.1	Declare an array with 7 decimal elements✓ Initialise first value as 15✓ Loop 6 times ✓ Assign value✓ to correct array element✓ Append value to output area✓ In correct format ✓ Increase next amount✓	**8**	
1.4.2	Randomly generate value✓ in correct range✓ Extract correct value from array✓ Display message with value included✓	**4**	
	TOTAL:	**48**	

ANNEXURE B:

SECTION B:

QUESTION 2: MARKING GRID – OBJECT-ORIENTATED PROGRAMMING

CENTRE NUMBER:	**EXAMINATION NUMBER:**		
QUESTION	**DESCRIPTION**	**MAX. MARKS**	**LEARNER'S MARKS**
2.1.1	**Constructor:** Correct constructor name and correct parameter ✓ Set the attribute to the parameter ✓ Set the number of members to 0 ✓ Set the topic list to empty string ✓	**4**	
2.1.2	**Accessor METHODS:** getForumName() ✓ getNumMembers() ✓ getTopicList() ✓	**3**	
2.1.3	**addMembers METHOD:** Correct method heading with parameter✓✓ Increment number of members attribute with parameter value✓	**3**	
2.1.4	**removeMembers METHOD:** Correct method heading with parameter✓✓ Decrement numMembers attribute by parameter value✓	**3**	
2.1.5	**addTopic METHOD:** Correct method heading, receive topic as a parameter ✓ If topic is not part of the topic list ✓ ✓ Join the topic to the topic list ✓ in a new line ✓	**5**	
2.1.6	**determinePopularityLevel METHOD:** Correct method heading ✓ Low range✓✓ Medium range ✓✓ High range ✓✓ Return level ✓	**8**	

QUESTION 2: MARKING GRID – OBJECT-ORIENTATED PROGRAMMING (continued)

	GUI		
2.2.1	Object class name ✓ object name ✓	**2**	
2.2.2	Extract forum name from combo box ✓ Instantiate forum object ✓ using the correct object name ✓ and parameter ✓ Place the forum name in the output box ✓ Place the Member count in the output box ✓	**6**	
2.2.3	(a) **[Add button]** Extract the number from the text box ✓ Call the addMembers method ✓ Display the updated value in the text box ✓ (b) **[Remove button]** Extract the number from the text box ✓ Test if the number is > number of members in the object ✓ Display message ✓ else Call the removeMembers method ✓ Display the updated value in text box ✓	**3** **5**	
2.2.4	Error handling technique ✓✓ Open the text file with the correct name ✓✓ Loop through the file ✓ Read a line ✓ Extract the forum name ✓ Extract the topic ✓ Check the forum name in the line with the object forum name ✓✓ Call the addTopic method ✓ with parameter ✓ Display a message stating the topic list is created✓	**13**	
2.2.5	Display the topic list✓ in the output area provided with the getTopic method✓	**2**	
2.2.6	Use the object to call the determinePopulalityLevel method✓ Display data in the output area provided✓	**2**	
	TOTAL:	**59**	

ANNEXURE C:

SECTION C:

QUESTION 3: MARKING GRID – PROBLEM-SOLVING PROGRAMMING

CENTRE NUMBER:	**EXAMINATION NUMBER:**		
QUESTION	**DESCRIPTION**	**MAX. MARKS**	**LEARNER'S MARKS**
3.1	Initialize global variables ✓ **Button Rhino/Blue Crane:** Set conservation type to Blue Crane/Rhino✓ Set conservation code to B or R✓ Set button not selected invisible✓ change heading to conservation type✓ loop through pledges in the array✓ test for conservation code✓ calculate and display total amount for selected project✓ calculate and display number of pledges for selected project✓ calculate ✓ and display percentage in correct format✓ If the percentage < 40✓, change panel background to red✓ else change panel background to yellow✓	**14**	
3.2	Loop through arrays✓ If the conservation code in the array ✓matches the selected project✓, Display the amount pledged✓	**4**	
3.3	Read new amount ✓ Test if the conservation code is not empty✓ AND test if entries is less than 20✓ Add value to arrAmounts array✓ Add B/R to arrConservationType array in corresponding index✓ Add value to total pledged for selected project ✓ Increment total pledges ✓ Increment number of pledges for selected project✓ Display message indicating pledge made✓ If not valid pledge display message✓ Update the GUI with new values for: • total amount pledged✓, • total number of pledges✓ and • percentage pledged✓.	**13**	

QUESTION 3: MARKING GRID – PROBLEM-SOLVING PROGRAMMING (continued)

3.4	**Remove low pledges:** Create a counter to start at the first position of array✓ Loop- while not found✓ & counter less than number of pledges✓ Test✓ if the pledge value < 100✓ Loop from index to last entry✓ Overwrite amount✓ and conservation type✓ in array with next value✓ Decrement total pledges✓ Increment counter✓ Update the GUI with new values for: • total amount pledged, • total number of pledges and • percentage pledged✓.	**12**	
	TOTAL:	**43**	

SUMMARY OF LEARNER'S MARKS:

	QUESTION 1	QUESTION 2	QUESTION 3	GRAND TOTAL
MAX. MARKS	**48**	**59**	**43**	**150**
LEARNER'S MARKS				

ANNEXURE D: SOLUTION FOR QUESTION 1: JAVA

```
================================================================================
// Question 1.1
================================================================================

private void btnTandCActionPerformed(java.awt.event.ActionEvent evt) {
     if (chbTerms.isSelected()) {
              chbTerms.setForeground(Color.green);
              btnTandC.setText("Continue to next section");
        } else {
           chbTerms.setForeground(Color.red);
        }
}
================================================================================
// Question 1.2
================================================================================

private void btnSubmitRegActionPerformed(java.awt.event.ActionEvent evt) {
        String userName = txfUname.getText().trim();
        String password = txfPassword.getText();
        boolean letters = false, numbers = false;
        for (int i = 0; i < password.length(); i++) {
            if (Character.isLetter(password.charAt(i))) {
                letters = true;
            }
            if (Character.isDigit(password.charAt(i))) {
                numbers = true;
            }
        }
     if (letters && numbers && password.length() >= 8 &&
         password.indexOf(" ") == -1) {
            btnSubmitReg.setEnabled(false);
            picture.setVisible(true);
        } else {
            JOptionPane.showMessageDialog(null, "Enter a valid password");
            txfPassword.setText("");
      }
}
================================================================================
// Question 1.3.1
================================================================================
private void btnMessageDetailsActionPerformed(java.awt.event.ActionEvent evt) {

        Calendar today = Calendar.getInstance();
        cDate = String.format("20%ty-%tm-%td", today, today, today);
        email = (String) (cbxSendEmail.getSelectedItem());
        String message = txaMessage.getText();
        txaMessage.setText("\n" + cDate+"\n" + email +"\n\n" +message);
    }
================================================================================
// Question 1.3.2
================================================================================
private void btnWriteToFileActionPerformed(java.awt.event.ActionEvent evt) {
 try {
   PrintWriter out = new PrintWriter(new FileWriter("EmailRecords.txt", true));
   out.println(cDate + "#" + email);
   JOptionPane.showMessageDialog(null, "Information written to file");
   out.close();
  } catch (Exception e) {
 }
}
```

```
================================================================================
// Question 1.4.1
================================================================================

private void btnPopulateArrayActionPerformed(java.awt.event.ActionEvent evt) {

        double amount = 15;
        txaAirtime.setText("Airtime:\n");
        for (int cnt = 0; cnt < 7; cnt++) {
            airtime[cnt] = amount;
            txaAirtime.append(String.format("%-8.2f%n", amount));
            amount = amount * 1.5;
        }
    }
================================================================================
// Question 1.4.2
================================================================================

private void btnWinningActionPerformed(java.awt.event.ActionEvent evt) {
        int num = (int)(Math.random() * 7);
        JOptionPane.showMessageDialog(rootPane, "You won " + String.format
        ("R%-6.2f",airtime[num])+" of airtime!");
}
```

ANNEXURE E: SOLUTION FOR QUESTION 2: JAVA

```
//    A solution to Question 2
```

OBJECT CLASS: FORUM (GIVEN)

```
//============================================================================
//This code is given in the program
//============================================================================
    private String forumName;
    private int numMembers;
    private String topicList;


public void clearTopicList(){
  topicList = "";
}

//============================================================================
//New Code

//============================================================================
// Question 2.1.1   Constructor method
//============================================================================

     public Forum(String forumName) {
        this.forumName = forumName;
        this.numMembers = 0;
        topicList = "";
    }

//============================================================================
// Question 2.1.2  Three accessor methods
//============================================================================

  public String getForumName(){
         return forumName;
     }

         public int getNumMembers(){
        return numMembers;
    }

         public String getTopicList(){
        return topicList;
    }
============================================================================
// Question 2.1.3   addMembers method
============================================================================
    public void addMembers(int number) {
        numMembers += number;
    }
============================================================================
// Question 2.1.4   removeMembers method
============================================================================

    public void removeMembers(int number) {
        numMembers -= number;
    }
```

```
================================================================================
// Question 2.1.5   addTopic method
================================================================================

       public void addTopic(String topic) {
        if (topicList.indexOf(topic) == -1) {
            topicList = topicList + topic + "\n";
        }
    }
================================================================================
// Question 2.1.6     determinePopularityLevel method
================================================================================

      public String determinePopularityLevel(){
        String level = "Low";
        if(numMembers>200 && numMembers <= 500)
            level = "Medium";
             if(numMembers>500 )
            level = "High";
        return level;
    }
```

GUI CLASS: QUESTION2GUI

```
================================================================================
// Question 2.2.1     Instantiate an object of the Forum class
================================================================================

 Forum forum;

================================================================================
// Question 2.2.2       Create forum object button
================================================================================

  private void btnCreateObjectActionPerformed(java.awt.event.ActionEvent evt) {
        String forumName = (String) (cbxForumNames.getSelectedItem());
        forum = new Forum(forumName);
        txfForumName.setText(forum.getForumName());
        txfNumMembers.setText(""+forum.getNumMembers());
    }
================================================================================
// Question 2.2.3       Add and Remove numMembers button
================================================================================
  // Add Button

        private void btnAddMemActionPerformed(java.awt.event.ActionEvent evt) {
        int addNum = Integer.parseInt(txfUpdateMembers.getText());
        forum.addMembers(addNum);
        txfNumMembers.setText("" + forum.getNumMembers());
    }
 // Remove Button
     private void btnRemoveMemActionPerformed(java.awt.event.ActionEvent evt) {
        int delNum = Integer.parseInt(txfUpdateMembers.getText());
         if(delNum > forum.getNumMembers())
          JOptionPane.showMessageDialog(null,"Invalid number of members to be
          deleted");
         else{
        forum.removeMembers((delNum));
        txfNumMembers.setText("" + forum.getNumMembers());
         }
    }
```

```
================================================================================
// Question 2.2.4          Create topic list button
================================================================================
private void btnCreateTopicListActionPerformed(java.awt.event.ActionEvent evt)
{
        try {
            Scanner sc = new Scanner(new FileReader("DataQ2.txt"));
            while (sc.hasNext()) {
                String line = sc.nextLine();
                Scanner d = new Scanner(line).useDelimiter("#");
                String forumName = d.next();
                String topic = d.next();
                if (forumName.equalsIgnoreCase(forum.getForumName())) {
                    forum.addTopic(topic);
                }
                txaOutput.setText("Topic list created successfully");
            }
            sc.close();
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, "The file does not exist");
        }
    }
================================================================================
// Question 2.2.5          Display topic list for selected forum
================================================================================

   private void btnDisplayTopicsActionPerformed(java.awt.event.ActionEvent evt)
{
        txaOutput.setText(forum.getTopicList());
    }

================================================================================
// Question 2.2.6          Determine the popularity level of the forum
================================================================================

 private void btnPopularityLevelActionPerformed(java.awt.event.ActionEvent evt)
{
     txaOutput.setText("POPULARITY LEVEL:"+forum.determinePopularityLevel());
```

ANNEXURE F: SOLUTION FOR QUESTION 3: JAVA

```
================================================================================
// Question 3 – Class PopulateArrays
================================================================================

//==============================================================================
//This code is given
//==============================================================================

package Question3Package;


public class PopulateArrays {

    public char[] populateConservationType() {
        char[] arrConservationType = new char[100];
        arrConservationType[0] = 'B';
        arrConservationType[1] = 'B';
        arrConservationType[2] = 'R';
        arrConservationType[3] = 'B';
        arrConservationType[4] = 'R';
        arrConservationType[5] = 'B';
        arrConservationType[6] = 'R';
        arrConservationType[7] = 'B';
        arrConservationType[8] = 'R';
        arrConservationType[9] = 'B';
        arrConservationType[10] = 'B';
        return arrConservationType;
    }

     public double[] populateAmountArray() {
        double[] arrAmounts = new double[100];
        arrAmounts[0] = 3299;
        arrAmounts[1] = 1216;
        arrAmounts[2] = 928;
        arrAmounts[3] = 456;
        arrAmounts[4] = 996;
        arrAmounts[5] = 3753;
        arrAmounts[6] = 60;
        arrAmounts[7] = 585;
        arrAmounts[8] = 456;
        arrAmounts[9] = 4354;
        arrAmounts[10] = 95;
        return arrAmounts;
    }


}
```

```
//============================================================================
// Question 3 – Class Question3 – GUI Class
//============================================================================
package Question3Package;

import java.awt.Transparency;
import java.awt.color.ColorSpace;
import java.io.FileNotFoundException;
import java.text.DecimalFormat;
import javax.swing.JOptionPane;

public class Question3MEMO_GUI extends javax.swing.JFrame {

//============================================================================
//This code is given in the program
//============================================================================

    PopulateArrays objAssign = new PopulateArrays();
    int totalPledges = 11;
    int cntProjectPledges = 0;
    double ultimateGoal = 125000.00;
    double[] arrAmounts = new double[20];
    char[] arrConservationType = new char[20];

//============================================================================
//This code is given in the program under the constructor
//============================================================================

       public Question3_Memo() {
        initComponents();
        this.setLocationRelativeTo(this);
        lblUltimateGoalAmount.setText("" + ultimateGoal);
        arrConservationType = objAssign.populateConservationType();
        arrAmounts = objAssign.populateAmountArray();
        lblUltimateGoalAmount.setText("R 125000.00");
        lblTotalAmountPledged.setText("R 0.00");
        lblNumPledges.setText("0");
        lblPercentage.setText("%");
    }


//============================================================================
// New code
//============================================================================
    double projectTotal = 0;
    char conservationCode = ' ';
    String conservationType = "";

//============================================================================
// Code for graphical interface not copied
//============================================================================

//============================================================================
// Events on buttons for conservation projects – 3.1
//============================================================================
    private void btnRhinoActionPerformed(java.awt.event.ActionEvent evt) {
        conservationCode = 'R';
        conservationType = "Rhino";
        btnBlueCrane.setVisible(false);
        process();    }
```

```
    private void btnBlueCraneActionPerformed(java.awt.event.ActionEvent evt) {
        conservationCode = 'B';
        conservationType = "Blue crane";
        btnRhino.setVisible(false);
        process();    }
// Method used in 3.1
     public void process() {
        projectTotal = 0;
        cntProjectPledges = 0;
        lblConservationtype.setText(conservationType + " conservation");
        for (int cnt = 0; cnt < totalPledges; cnt++) {
            if (arrConservationType[cnt] == conservationCode) {
                projectTotal += arrAmounts[cnt];
                cntProjectPledges++;
            }
            lblTotalAmountPledged.setText(String.format("R%-7.2f",
            projectTotal));
            double percent = (double)projectTotal / ultimateGoal * 100;
            System.out.println(percent);
            lblPercentage.setText(String.format("%-7.2f%-1s", percent,"%"));
            if(percent<40)
            {
                pnlProject.setBackground(Color.red);
            }
            else
            {
                pnlProject.setBackground(Color.yellow);
            }
        }
        lblNumPledges.setText("" + cntProjectPledges);
    }
//=========================================================================
// Events on pledges for selected project button - 3.2
//=========================================================================
private void
btnPledgesForSelectedProjectActionPerformed(java.awt.event.ActionEvent evt) {
        txaOutputAll.setText("");
        for (int cnt = 0; cnt < totalPledges; cnt++) {
            if (arrConservationType[cnt] == conservationCode) {
                txaOutputAll.append(String.format("R%7.2f%n",
                arrAmounts[cnt]));
            }
        }   }
//=========================================================================
// Events on make new pledge button - 3.3
//=========================================================================
private void btnNewPledgeActionPerformed(java.awt.event.ActionEvent evt) {
        double amount = Double.parseDouble(txfAmount.getText());
        if (totalPledges < 20 && amount > 0 && conservationCode != ' ') {
            arrAmounts[totalPledges] = amount;
            arrConservationType[totalPledges] = conservationCode;
            totalPledges++;
            cntProjectPledges++;
            projectTotal = projectTotal + amount;
            JOptionPane.showMessageDialog(null, "Your pledge has been
            registered");
        } else {
            JOptionPane.showMessageDialog(null, "Pledge has been
            unsuccessful");
     }
        txfAmount.setText("");
        process();
    }
```

```
//============================================================================
// Events on remove all low pledges button - 3.4
//============================================================================

private void btnRemoveLowPledgesActionPerformed(java.awt.event.ActionEvent evt)
{

        int index = 0;
        while (index < totalPledges) {

            if (arrAmounts[index] < 100) {
                for (int i = index; i < totalPledges - 1; i++) {
                    arrAmounts[i] = arrAmounts[i + 1];
                    arrConservationType[i] = arrConservationType[i + 1];
                }
                totalPledges--;
                index--;
            }
            index++;
        }
        process();
    }
```

ANNEXURE G: SOLUTION FOR QUESTION 1: DELPHI

```
unit Question1_U_Memo;
   //Solution for question 1.

interface

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

type
  TfrmQuestion1 = class(TForm)
    grpOptionA: TGroupBox;
    grpOptionB: TGroupBox;
    grpOptionC: TGroupBox;
    grpOptionD: TGroupBox;
    redTandC: TRichEdit;
    chkTerms: TCheckBox;
    btnSubmit: TButton;
    Label1: TLabel;
    lblUsername: TLabel;
    lblPassword: TLabel;
    edtUsername: TEdit;
    edtPassword: TEdit;
    btnSubmitLogin: TButton;
    bmbClose: TBitBtn;
    btnSendMessage: TButton;
    redMessage: TRichEdit;
    cboSendMail: TComboBox;
    imgPicture: TImage;
    btnWriteToFile: TButton;
    redAirtime: TRichEdit;
    btnPopulateArray: TButton;
    btnWinning: TButton;
    Label2: TLabel;
    procedure btnSubmitClick(Sender: TObject);
    procedure btnSubmitLoginClick(Sender: TObject);
    procedure btnSendMessageClick(Sender: TObject);
    procedure btnPopulateArrayClick(Sender: TObject);
    procedure btnWinningClick(Sender: TObject);
    procedure btnWriteToFileClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuestion1: TfrmQuestion1;

  sDate, sEmail : string;
  arrAirTime : array[1..7] of Real;

implementation

{$R *.dfm}
```

```
{$R+}
procedure TfrmQuestion1.btnSubmitClick(Sender: TObject);
begin
  //Code Question 1.1.
  if chkTerms.Checked
   then
   begin
      btnSubmit.Caption := 'Continue to next section';
      chkTerms.Font.Color := clGreen;
   end
   else chkTerms.Font.Color := clRed;
end;

procedure TfrmQuestion1.btnSubmitLoginClick(Sender: TObject);
var
  sUserName, sPassword : String;
  sMessage : String;
  bLetter, bDigit, bChar : Boolean;
  A : integer;
begin
 //Code Question 1.2.
  sUserName := Trim(edtUsername.Text);
  sPassword := Trim(edtPassword.Text);
  bLetter := False;
  bDigit := False;
  bChar := False;
  for A := 1 to Length(sPassword) do
  begin
    if UpCase(sPassword[A]) in ['A'..'Z']
     then bLetter := true;

    if sPassword[A] in ['0'..'9']
     then bDigit := true;
  end; // for
  if (Length(sPassword) < 8) OR (bLetter = False) OR (bDigit = false)
                                              or (Pos(' ',sPassword) > 0)
  then
  begin
    MessageDlg('Enter a valid password', mtError, [mbOK], 0);
    edtPassword.Clear;
    edtPassword.SetFocus;
  end
  else
  begin
      btnSubmitLogin.Enabled := False;
      imgPicture.Visible := True;
  end;
end;
procedure TfrmQuestion1.btnSendMessageClick(Sender: TObject);
var
  sLine : string;
begin
//Question 1.3.1.
  sDate    := FormatDateTime('yyyy-mm-dd', Now());
  sEmail := cboSendMail.Items[cboSendMail.ItemIndex];
  sLine :=  sDate + #13 + sEmail + #13 +
            #13 + redMessage.Text + #13 ;
  redMessage.Text := sLine;
end;
```

```
procedure TfrmQuestion1.btnWriteToFileClick(Sender: TObject);
var
   txtFile : TextFile;
begin
   //Question 1.3.2.
   AssignFile(txtFile, 'EmailRecords.txt');
   Append(txtFile);
   Writeln(txtFile, sDate + '#' + sEmail);
   CloseFile(txtFile);
   MessageDlg('Information written to file.', mtInformation, [mbok],0);
end;

procedure TfrmQuestion1.btnPopulateArrayClick(Sender: TObject);
var
  A  : Integer;
begin
  //Question 1.4.1
  arrAirTime[1] := 15;
  for A := 2 to 7 do
    arrAirTime[A] := arrAirTime[A-1] * 1.5;

  redAirtime.Clear;
  redAirtime.Lines.Add('Airtime:' + #13);
  for A := 1 to 7 do
   redAirtime.Lines.Add(FloatToStrF(arrAirTime[A], ffFixed, 8,2));
end;

procedure TfrmQuestion1.btnWinningClick(Sender: TObject);
var
  iNum : Integer;
  sMsg : string;
begin
  //Question 1.4.2
  Randomize;
  iNum := Random(7)+1;
  sMsg := 'You won ' + FloatToStrF(arrAirTime[iNum], ffCurrency, 8, 2) +
          ' of airtime!';
  MessageDlg(sMsg, mtInformation, [mbOK], 0);
end;

procedure TfrmQuestion1.FormCreate(Sender: TObject);
begin
     sDate  := '';
     sEmail := '';
     CurrencyString := 'R';
end;

end.
```

ANNEXURE H: SOLUTION FOR QUESTION 2: DELPHI

2.1 FORUM CLASS UNIT

```
unit Forum_U;
 //Solution for Question 2 - class unit.

interface

type
  TForum = class(TObject)
     private
       fForumName  : string;
       fNumMembers : Integer;
       fTopicList  : string;

     public
        constructor Create(sForumName : String);
       function getForumName : string;
       function getNumberMembers : Integer;
       function getTopicList : string;
       procedure addMembers(iNumber : Integer);
       procedure removemembers(iNumber : Integer);
       procedure addTopic(sNewTopic : String);
       function determinePopularityLevel : String;
       procedure clearTopicList;
  end;

implementation

uses SysUtils;

{ TForum }

// Question 2.1.1
constructor TForum.Create(sForumName: String);
begin
   fForumName  := sForumName;
   fNumMembers := 0;
   fTopicList := '';
end;
// Question 2.1.6
function TForum.determinePopularityLevel: String;
begin
  case fNumMembers of
    0..200   : result := 'Low';
    201..500 : result := 'Medium';
    else
      result := 'High';
  end;
end;
// Question 2.1.2
function TForum.getForumName: string;
begin
   result := fForumName;
end;
```

```
function TForum.getNumberMembers: Integer;
begin
  Result := fNumMembers;
end;

function TForum.getTopicList: string;
begin
  result := fTopicList;
end;
// Question 2.1.3
procedure TForum.addMembers(iNumber: Integer);
begin
  Inc(fNumMembers, iNumber);
end;
// Question 2.1.4
procedure TForum.removeMembers(iNumber: Integer);
begin
  Dec(fNumMembers, iNumber);
end;
// Question 2.1.5
procedure TForum.addTopic(sNewTopic: String);
begin
  if Pos(sNewTopic, fTopicList) = 0
   then fTopicList := fTopicList + sNewTopic + #13;
end;

procedure TForum.clearTopicList;
begin
  fTopicList := '';
end;

end.
```

2.2 MAIN FORM UNIT

```
unit Question2_U_Memo;
   //Solution for Question 2.
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, StdCtrls, Buttons, ExtCtrls, ComCtrls;
type
  TfrmQuestion2 = class(TForm)
    pnlBtns: TPanel;
    bmbClose: TBitBtn;
    grpQ222: TGroupBox;
    Label1: TLabel;
    grpQuest223: TGroupBox;
    cboSport: TComboBox;
    btnCreateObject: TButton;
    Label3: TLabel;
    Label4: TLabel;
    edtUpdateMembers: TEdit;
    btnAddMem: TButton;
    btnRemoveMember: TButton;
    grpQ224_226: TGroupBox;
    Label5: TLabel;
    btnCreateTopicList: TButton;
```

```
    btnDefaultTopicList: TButton;
    btnPopularity: TButton;
    btnDisplayTopicList: TButton;
    redQ2: TRichEdit;
    Label6: TLabel;
    edtForumName: TEdit;
    Label7: TLabel;
    edtMemberCount: TEdit;
    Label2: TLabel;
    procedure btnCreateObjectClick(Sender: TObject);
    procedure btnAddMemClick(Sender: TObject);
    procedure btnRemoveMemberClick(Sender: TObject);
    procedure btnCreateTopicListClick(Sender: TObject);
    procedure btnDefaultTopicListClick(Sender: TObject);
    procedure btnDisplayTopicListClick(Sender: TObject);
    procedure btnPopularityClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuestion2: TfrmQuestion2;

implementation

uses
  Forum_U;

var
  Forum : TForum;
// Question 2.2.1

{$R *.dfm}
{$R+}

procedure TfrmQuestion2.btnCreateObjectClick(Sender: TObject);
var
  sForumName : String;
  iNumber : integer;
begin
//Question 2.2.2
  sForumName := cboSport.Items[cboSport.ItemIndex];
  Forum := TForum.Create(sForumName);
  edtForumName.Text := Forum.getForumName;
  edtMemberCount.Text := IntToStr(Forum.getNumberMembers);
end;

procedure TfrmQuestion2.btnAddMemClick(Sender: TObject);
var
  iNumber : Integer;
begin
//Question 2.2.3 - Add.
  iNumber := StrToInt(edtUpdateMembers.Text);
  Forum.addMembers(iNumber);
  edtForumName.Text := Forum.getForumName;
  edtMemberCount.Text := IntToStr(Forum.getNumberMembers);
end;
```

```
procedure TfrmQuestion2.btnRemoveMemberClick(Sender: TObject);
var
  iNumber : Integer;
begin
//Question 2.2.3 - Remove.
  iNumber := StrToInt(edtUpdateMembers.Text);
  if iNumber <= Forum.getNumberMembers then
    Forum.removeMembers(iNumber)
    else
    MessageDlg('Invalid number of members to be removed',mtError,[mbOK],0);
  edtMemberCount.Text := IntToStr(Forum.getNumberMembers);
end;

procedure TfrmQuestion2.btnCreateTopicListClick(Sender: TObject);
var
  txtFile : TextFile;
  sLine   : String;
begin
//Question 2.2.4 – Create topic list
  if NOT FileExists('DataQ2.txt')
   then
    begin
      MessageDlg('File does not exists', mtError, [mbOK], 0);
      Exit;
    end;//if
  AssignFile(txtFile, 'DataQ2.txt');
  Reset(txtFile);
  While NOT EOF(txtFile) do
   begin
      Readln(txtFile, sLine);
      if Pos(UpperCase(Forum.getForumName), UpperCase(sLine)) = 1
       then
        begin
          Delete(sLine, 1, Pos('#', sLine));
          Forum.addTopic(sLine);
        end;
   end;//while
  CloseFile(txtFile);
  //redQ2.Lines.Add('Topic list created successfully');
  redQ2.Lines.SetText('Topic list created successfully');
end;

procedure TfrmQuestion2.btnDefaultTopicListClick(Sender: TObject);
var
  sTemp : string;
begin
//Question 2.2.4 - Bypass
  forum.clearTopicList;
  sTemp := 'Hip Hop,Composing Lyrics,Music classes,Concerts,Musical
instruments'
           + ',Talk shows,Live shows,Theatre and drama';
  forum.addTopic(sTemp);
end;
```

```
procedure TfrmQuestion2.btnDisplayTopicListClick(Sender: TObject);
var
  i, p : Integer;
  topics : string;
begin
//Question 2.2.5
  redQ2.Clear;
  redQ2.Lines.Add(Forum.getTopicList);
end;

procedure TfrmQuestion2.btnPopularityClick(Sender: TObject);
begin
// Question 2.2.6
  redQ2.Clear;
  redQ2.Lines.Add('POPULARITY LEVEL: ' + Forum.determinePopularityLevel);
end;

end.
```

ANNEXURE I: SOLUTION FOR QUESTION 3: DELPHI

```
unit Question3_U_Memo;
    //Enter your examination number here.

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, JPeg, ImgList, ExtCtrls, StdCtrls, Buttons, ComCtrls;

type
  TfrmQuestion3 = class(TForm)
    pnlHeadings: TPanel;
    lblConservationType: TLabel;
    bmbCrane: TBitBtn;
    bmbRhino: TBitBtn;
    pnlHeadingTotals: TPanel;
    grpNewPledge: TGroupBox;
    grpList: TGroupBox;
    Label5: TLabel;
    edtAmount: TEdit;
    btnNewPledge: TButton;
    redAll: TRichEdit;
    btnDisplayAll: TButton;
    lblUGoal: TLabel;
    lblTAP: TLabel;
    lblTP: TLabel;
    lblUltimateGoal: TLabel;
    lblTotalAmount: TLabel;
    Label1: TLabel;
    Label2: TLabel;
    lblProjectPledges: TLabel;
    lblPercentage: TLabel;
    GroupBox1: TGroupBox;
    btn1: TButton;
    btnChangeSelection: TButton;
    procedure bmbCraneClick(Sender: TObject);
    procedure bmbRhinoClick(Sender: TObject);
    procedure btnNewPledgeClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure btnDisplayAllClick(Sender: TObject);
    procedure btnChangeSelectionClick(Sender: TObject);
    procedure process;
    procedure btnRemoveLowPledgesClick(Sender: TObject);
  private
    { Private declarations }
    procedure SetupArrays;
  public
    { Public declarations }
  end;

var
  frmQuestion3: TfrmQuestion3;

implementation
{$R *.dfm}
{$R+}
```

```
var
  arrConservationType       : array[1..20] of char;
  arrAmounts                : array[1..20] of real;
  cSelection                : Char;
  sConservationType         : string;
  iNumPledges               : Integer = 11;
  iProjectPledges           : Integer  = 0;
  rTotalAmount              : Real = 0.00;
  rUltimateGoal             : Real = 125000.00;

procedure TfrmQuestion3.bmbCraneClick(Sender: TObject);
begin
  //Code Question 3.1. - Crane.
  cSelection := 'B';
  sConservationType := 'Blue Crane';
  bmbRhino.Hide;
  process;
end;

procedure TfrmQuestion3.bmbRhinoClick(Sender: TObject);
begin
  //Code Question 3.1. - Rhino.
  cSelection := 'R';
  sConservationType := 'Rhino';
  bmbCrane.Hide;
  process;
end;

procedure TfrmQuestion3.btnDisplayAllClick(Sender: TObject);
var
  i : Integer;
begin
  //Question 3.2.
  redAll.Clear;
  for i := 1 to iNumPledges do
    if cSelection = arrConservationType[i] then
     redAll.Lines.Add(FloatToStrF(arrAmounts[i],ffCurrency,8,2));
end;

procedure TfrmQuestion3.btnNewPledgeClick(Sender: TObject);
var
  rAmnt : real;
begin
 //Question 3.3.
 rAmnt := StrToFloat(edtAmount.Text);
 if (iNumPledges < 20) and (rAmnt > 0) and (cSelection <> ' ')
  then
   begin
    Inc(iNumPledges);
    arrAmounts[iNumPledges] := rAmnt;
    arrConservationType[iNumPledges] := cSelection;
    showMessage('Pledge has been registered')
   end
  else
     ShowMessage('Pledge has been unsuccessful');
   edtAmount.Text := '0';
   process;
end;
```

```
procedure TfrmQuestion3.btnRemoveLowPledgesClick(Sender: TObject);
var
  i, indx : Integer;
begin
    //Question 3.5.
    indx := 0;
    while indx < iNumPledges do
    begin
      Inc(indx);
      if arrAmounts[indx] < 100
       then
       begin
        for i := indx to iNumPledges-1 do
         begin
           arrAmounts[i] := arrAmounts[i+1];
           arrConservationType[i] := arrConservationType[i+1];
         end;
         Dec(iNumPledges);
         Dec(indx);
       end;
    end;
    process;
end;

procedure TfrmQuestion3.FormCreate(Sender: TObject);
begin
  //data supplied:
  CurrencyString := 'R';
  SetupArrays;
  lblUltimateGoal.Caption := FloatToStrF(rUltimateGoal, ffCurrency, 8, 2);
  cSelection := ' ';
end;

procedure TfrmQuestion3.process;
var
   i : Integer;
   rPercentage : Real;
begin
  rTotalAmount := 0;
  iProjectPledges := 0;
  for i := 1 to iNumPledges do
     if (arrConservationType[i] = cSelection)
      then
      begin
         rTotalAmount := rTotalAmount + arrAmounts[i];
         iProjectPledges := iProjectPledges +1;
      end;
  lblConservationType.Caption :=   sConservationType + ' Conservation';
  lblUltimateGoal.Caption := FloatToStrF(rUltimateGoal, ffCurrency, 8, 2);
  lblTotalAmount.Caption := FloatToStrF(rTotalAmount,ffCurrency,8,2);
  lblProjectPledges.Caption := IntToStr(iProjectPledges);
  rPercentage := rTotalAmount / rUltimateGoal * 100;
  lblPercentage.Caption := FloatToStrF(rPercentage,ffFixed,6,2)+ '%';
  if  (rPercentage >= 40)
   then
    pnlHeadingTotals.Color := clYellow
   else
   pnlHeadingTotals.Color := clRed;
end;
```

```
procedure TfrmQuestion3.btnChangeSelectionClick(Sender: TObject);
begin
// Given code for reset button
  bmbRhino.Visible := true;
  bmbCrane.Visible := True;
  lblConservationType.Caption := 'Conservation';
  edtAmount.Text := '0';
  lblUltimateGoal.Caption := FloatToStrF(rUltimateGoal, ffCurrency, 8, 2);
  lblTotalAmount.Caption := 'R 0.00';
  lblProjectPledges.Caption := '0';
  lblPercentage.Caption := '0%';
  pnlHeadingTotals.Color := clBtnFace;
  cSelection := ' ';
  redAll.Clear;
end;

procedure TfrmQuestion3.SetupArrays;
begin
  //Given code to fill arrays
  arrAmounts[1]  := 3299;
  arrAmounts[2]  := 1216;
  arrAmounts[3]  := 928;
  arrAmounts[4]  := 456;
  arrAmounts[5]  := 996;
  arrAmounts[6]  := 3753;
  arrAmounts[7]  := 60;
  arrAmounts[8]  := 585;
  arrAmounts[9]  := 456;
  arrAmounts[10] := 4354;
  arrAmounts[11] := 95;

  arrConservationType[1]  := 'B';
  arrConservationType[2]  := 'B';
  arrConservationType[3]  := 'R';
  arrConservationType[4]  := 'B';
  arrConservationType[5]  := 'R';
  arrConservationType[6]  := 'B';
  arrConservationType[7]  := 'R';
  arrConservationType[8]  := 'B';
  arrConservationType[9]  := 'R';
  arrConservationType[10] := 'B';
  arrConservationType[11] := 'B';

end.
```