basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

NATIONAL SENIOR CERTIFICATE

GRADE 12

INFORMATION TECHNOLOGY P1

EXEMPLAR 2014

MEMORANDUM

MARKS: 150

This memorandum consists of 26 pages.

INSTRUCTIONS FOR THE MARKER

1. 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.

2. There may be different views about some matters of emphasis or detail in the guidelines and different interpretations of the application thereof.

3. Note that learners who provide an alternate correct solution to that given in the marking guidelines will be given full credit for the relevant answer if the instructions in the question paper were followed.

4. **ANNEXURES A, B** and **C** (pages 3–5) contain the marking grid for each question for using either one of the two programming languages.

5. **ANNEXURES D, E** and **F** (pages 6–17) contain the solutions for DELPHI for QUESTIONS 1 to 3 in programming code.

6. **ANNEXURES G, H** and **I** (pages 18–26) contain the solutions for JAVA for QUESTIONS 1 to 3 in programming code.

7. Copies of ANNEXURES A, B and C (pages 3–5) should be made for EACH learner and completed during the marking session.

ANNEXURE A:

QUESTION 1: MARKING GRID

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	**DESCRIPTION**	**MAX. MARKS**	**LEARNER'S MARKS**
1.1	Change the caption property✓✓	2	
1.2	Extract name ✓ and surname ✓ Extract ID ✓ Determine initials ✓✓✓ Use ID to determine value in correct range ✓ Determine If Male ✓ or Female ✓ Initial and surname uppercase✓ Compiled string✓ Display tag ✓	12	
1.3	Read ID✓ Extract first TWO<YY>characters from ID✓ Convert the year into an integer value ✓ If condition✓to add either 2000 ✓ or 1900 to year. ✓ Calculate age✓ Using case/switch/if with year value ✓ Option 1: age 14, 15 Assign 'u/15' to age group ✓ Option 2: age 16, 17 Assign 'u/17' to age group ✓ Option 3: age 18, 19 Assign 'u/19' to age group ✓ If not one of above – Display not eligible ✓ Concatenate and display age group to name tag at end ✓	13	
1.4	Input unit price ✓ and quantity ✓ from keyboard with InputBox/InputDialog Convert string values to floating point values✓ Display headings/subheadings ✓ in column format✓ Loop ✓using quantity✓ Calculate cost ✓ For every second item ✓ Calculate 20% discount ✓✓ Display labels and calculated values in columns ✓ with correct format ✓ Input amount tendered✓ Calculate change✓✓ Calculate in rands ✓ 50c, ✓ 20c, ✓10c ✓ Display change rands and coins ✓ Test if more than 0 ✓ Then display values formatted to currency with two decimals✓	23	
	TOTAL	**50**	

ANNEXURE B:

QUESTION 2: MARKING GRID

CENTRE NUMBER:	EXAMINATION NUMBER:		
QUESTION	**DESCRIPTION**	**MAX. MARKS**	**LEARNER'S MARKS**
2.1	Class name ✓ Private variables ✓with correct String data types ✓ Double data type✓	4	
2.1.1	Define constructor with four parameters ✓ Correctly assign different parameters to local attributes (three string) (one numerical) ✓	2	
2.1.2	Four **Accessor methods** public✓ getEvent✓ ; getTeamName✓ getYear ✓; getRecordTime ✓	5	
2.1.3	**checkForRecord method - void**✓ Receive two parameters ✓ Get year value ✓ from system's date function ✓ if: Test if new time✓ = current record time✓ Concatenate new team name to current team name separate with ;✓ Concatenate new year value to current year value separate with ;✓ Else Test newtime ✓ < current record time✓ Assign new team name ✓ Assign new year value ✓ Assign new record time ✓	13	
2.1.4	**toString method** Concatenate local attributes ✓ Add labels ✓ and new line ✓ Format floating point values to two decimals ✓	4	
2.2.1	Declare object✓ Instantiate new object using class✓; create method with correct parameters ✓ in the right order✓ Use toString method to display information ✓	5	
2.2.2	User input from GUI (team name and time)✓✓ Validate time entered Convert string input to numerical value ✓ Call createButton method✓ On Error ✓ Display message and exit event✓ Clear text box and text area ✓ Specifications for new button Set panel to opaque ✓; initialise new button ✓ Add button ✓to panel ✓; set text ✓ Set bounds ✓✓; set visible to true ✓ Functionality for new button Add action listener ✓ and action performed ✓ Call **checkForRecord** method✓ with two paramaters✓ Display info ✓ with toString method✓	21	
2.2.3	Check if 2013 ✓or ✓ 2014 ✓is part of the year Display ✓a suitable message Recent ✓ Old ✓	6	
	TOTAL	**60**	

ANNEXURE C:

QUESTION 3: MARKING GRID

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	**DESCRIPTION**	**MAX. MARKS**	**LEARNER'S MARKS**
3.1	Use **Results.txt text file**: Open file ✓to read from✓ Use a loop to read line of text from file✓ Read line from text file ✓ Extract abbreviation from line of text✓ Initialise array elements = 0✓✓ Test if✓ abbreviation = input from user✓ Determine the gender ✓ Determine the place achieved ✓ (copy/delete/pos/split)✓ Increase correct array element ✓ Increase the total number performances✓ Close the file ✓	19	
	Construct heading with correct school name ✓ Using loops✓, display array values ✓ in grid cells✓		
3.2	Determine average Set values in array position 3 to 0✓ Use Loops ✓✓ Calculate total ✓✓ Divide total by 3 ✓ Store average in correct position ✓	17	
	Display schools with averages/high score Determine highest score ✓✓✓ Display heading ✓ Use a loop ✓ Test if ✓ highest score = average ✓ Add a star to appropriate school ✓ Display school data ✓✓		
3.3	Create temporary location ✓ Swap row 1 with row 2 ✓✓ Execute Option B✓	4	
	TOTAL	**40**	

	QUESTION 1	QUESTION 2	QUESTION 3	TOTAL
MAX. MARKS	**50**	**60**	**40**	**150**
LEARNER'S MARKS				

ANNEXURE D: SOLUTION FOR QUESTION 1: DELPHI

```
unit Question1Unit;

interface

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

type
  TfrmQuestion1 = class(TForm)
    lblHeading: TLabel;
    grpBackground: TGroupBox;
    lblFirstName: TLabel;
    edtFirstName: TEdit;
    lblSurname: TLabel;
    edtSurname: TEdit;
    lblID: TLabel;
    edtID: TEdit;
    memNameTag: TMemo;
    grbOutput: TGroupBox;
    memBackground: TMemo;
    btnQuestion11: TButton;
    btnQuestion12: TButton;
    btnQuestion13: TButton;
    btnQuestion14: TButton;
    pnlImage: TPanel;
    imgAthlete: TImage;
    redOutput: TRichEdit;
    btnBack: TButton;
    procedure btnQuestion11Click(Sender: TObject);
    procedure btnQuestion12Click(Sender: TObject);
    procedure btnQuestion13Click(Sender: TObject);
    procedure btnQuestion14Click(Sender: TObject);
    procedure btnBackClick(Sender: TObject);
  private
    { Private declarations }
    sNameTag : String;
  public
    { Public declarations }
  end;

var
  frmQuestion1: TfrmQuestion1;

implementation

uses Math;

{$R *.dfm}
```

```
//*********************************************************************************************
//Question 1.1
//*********************************************************************************************

procedure TfrmQuestion1.btnQuestion11Click(Sender: TObject);
begin
  //Question 1.1
  lblHeading.Caption := 'PC Athletics Championships';
end;

//*********************************************************************************************
//Question 1.2
//*********************************************************************************************
procedure TfrmQuestion1.btnQuestion12Click(Sender: TObject);
var
  sGenderDigit, sGender, sInitials, sName, sSurname, sID : string;
  K : integer;
begin
  //Question 1.2
  sName := edtFirstName.text;
  sSurname := edtSurname.text;
  sID := edtID.text;
  sInitials := '' + sName[1];
  sGenderDigit := copy(edtID.Text,7,4);
  for K := 1 to length(sName) do
   begin
      if sName[K] = ' ' then
      begin
        sInitials := sInitials + copy(sName, K + 1 ,1);
      end;
   end;

  sGender := 'Female';
  if StrToInt(sGenderDigit) >= 5000 then
     sGender := 'Male';
  sNameTag := sSurname + ' ' + sInitials + '.';
  sNameTag := 'Athlete:' + #9 + UpperCase(sNameTag) + #13 + #9 +
sGender;
  redOutput.Lines.Add(sNameTag);
end;

//*********************************************************************************************
//Question 1.3
//*********************************************************************************************

procedure TfrmQuestion1.btnQuestion13Click(Sender: TObject);
var
  sAgeGroup       : string;
  iAge, iYearBorn : integer;
begin
  //Question 1.3
  iYearBorn := StrToInt(copy(edtID.Text,1,2));
  if iYearBorn <= 14 then
    iYearBorn :=  2000 + iYearBorn
  else
    iYearBorn := 1900 + iYearBorn;
  iAge := 2014 - iYearBorn;
```

```
  case iAge of
    13..15 : sAgeGroup := 'u/15';
    16..17 : sAgeGroup := 'u/17';
    18..19 : sAgeGroup := 'u/19';
  else
    sAgeGroup := 'Not eligible to participate';
  end;

  redOutput.Clear;
  redOutput.Lines.Add(sNameTag);
  redOutput.Lines.Add('Age group: ' + sAgeGroup);
end;

//*****************************************************************************************
//Question 1.4
//*****************************************************************************************

procedure TfrmQuestion1.btnQuestion14Click(Sender: TObject);
          // Question 1.4
var
  rPrice, rCost, rAmountTendered, rChange,rCentsD, rDiscount : real;
  iQuantity, K : integer;
  rands,coins50,coins20,coins10,cents : integer;
  centsD : real;
begin
  redOutput.Clear;
  rPrice := StrToFloat(InputBox('Price per unit','Please type unit
price: ','18.50'));
     iQuantity := StrToInt(InputBox('Quantity','Please type in the
quantity:  ','9'));
     redPurchases.lines.Add('Quantity' + '   Price'+ #9 + '     Total'+
#9 +'Discount'+ #9  +'Amount Due');

     for K := 1 to iQuantity do
          begin
               rCost := rPrice * K;
               if (K MOD 2) = 0 then
             rDiscount := rCost * 20/100;

      redPurchases.lines.Add(IntToStr(K) + '    X     ' +
FloatToStrF(rPrice, ffCurrency,8,2) + ' = ' +  FloatToStrF(rCost,
ffCurrency,8,2) + #9 + FloatToStrF(rDiscount, ffCurrency,8,2) + #9 + #9
+ FloatToStrF((rCost - rDiscount),ffCurrency,8,2 ));
          end;
    rCost := rCost - rDiscount;
    rAmountTendered :=  StrToFloat(InputBox('Amount Tendered','Enter the
amount tendered','300'));
    rChange := rAmountTendered - rCost;

    coins10 := 0;
    rands := trunc(rChange);
    centsD := round((rChange - rands) * 100);
    cents := trunc(centsD);
    coins50 := trunc(cents / 50);
    cents := cents MOD 50;
    coins20 := trunc(cents / 20);
    cents := cents MOD 20;
    coins10 := trunc(cents / 10);
```

```
    cents := cents MOD 10;
    if (cents > 0) then
    coins10 := coins10 + 1;
    redPurchases.lines.Add(' ');
    redPurchases.lines.Add('Change : ' + #9 + FloatToStrF(rChange,
ffCurrency,8,2));
    if (rands > 0) then
     redPurchases.lines.Add('Rands : ' + #9 + FloatToStr(rands));

    if (coins50 > 0)then
          redPurchases.lines.Add('50c coins : ' + #9 +
FloatToStr(coins50));

    if (coins20 > 0)then
                redPurchases.lines.Add('20c coins : ' + #9 +
FloatToStr(coins20));

    if (coins10 > 0)then
                redPurchases.lines.Add('10c coins : ' + #9 +
FloatToStr(coins10));
end;

procedure TfrmQuestion1.btnBackClick(Sender: TObject);
begin
  Close;
end;

end.
```

ANNEXURE E: SOLUTION FOR QUESTION 2: DELPHI

```
//*************************************************************************************************
//Question 2.1 RelayEvent object class:
//*************************************************************************************************

unit clsRelayEvent_u;

interface

type
   TRelayEvent = class(TObject)
   private
        fEvent : String;
        fTeam  : String;
        fYear  : String;
        fRecordTime : real;
   public
        constructor Create(sEvent, sTeam, sYear : String; rRecTime :
real);
        function getTeam  : String;
        function getEvent : String;
        function getYear  : String;
        function getRecordTime : real;
        procedure checkForRecord(sNewTeam : String; rNewTime : real);
        function toString : String;
end;

implementation
uses SysUtils, DateUtils;

{ TRelayItem }

constructor TRelayEvent.Create(sEvent, sTeam, sYear: String; rRecTime:
real);
begin
   fEvent   := sEvent;
   fTeam    := sTeam;
   fYear    := sYear;
   fRecordTime := rRecTime;
end;

procedure TRelayEvent.checkForRecord(sNewTeam: String; rNewTime: real);
begin
   if rNewTime < fRecordTime then
     begin
       fTeam  := sNewTeam;
       fYear  := IntToStr(YearOf(Today()));
       fRecordTime := rNewTime;
     end
    else
     if rNewTime = fRecordTime then
       begin
         fTeam := fTeam + '; ' + sNewTeam;
         fYear  := fYear + '; ' + IntToStr(YearOf(Today()));
         //record time does not change
       end;
end;
```

```
function TRelayEvent.getRecordTime: real;
begin
     Result := fRecordTime;
end;

function TRelayEvent.getTeam: String;
begin
   result := fTeam;
end;

function TRelayEvent.getYear: String;
begin
   result := fYear;
end;

function TRelayEvent.toString: String;
begin
   Result := 'Current record information for ' + fEvent + #13 + #13+
             'Team: ' + fTeam + #13 +
             'Year: ' + fYear + #13 +
             'Time: ' + FloatToStrF(fRecordTime, ffFixed, 5,2) + '
seconds';
end;
end.
```

2.2 MAIN FORM

```
//*************************************************************************************************
//Question 2.2 Driver class:
//*************************************************************************************************
unit Question2Unit;

interface

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

type
  TfrmQuestion2 = class(TForm)
    grpHeading: TGroupBox;
    grpQ21: TGroupBox;
    grpQ23: TGroupBox;
    grpQ22: TGroupBox;
    grpOutput: TGroupBox;
    btnDisplayCurrentHolder: TButton;
    btnRecordStatus: TButton;
    btnValidateTime: TButton;
    edtTeam: TEdit;

    edtTime: TEdit;
    lblTeam: TLabel;
    lblTime: TLabel;
    lblHeading: TLabel;
    redOutput: TRichEdit;
    btnClose: TButton;
    procedure btnDisplayCurrentHolderClick(Sender: TObject);
```

```
    procedure btnRecordStatusClick(Sender: TObject);
    procedure btnValidateTimeClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure btnCheckStatus(Sender:TObject);
    procedure btnCloseClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuestion2: TfrmQuestion2;
  Boys19Relay : TRelayEvent;

implementation

{$R *.dfm}
procedure TfrmQuestion2.FormCreate(Sender: TObject);
begin
   Boys19Relay := TRelayEvent.Create('4x100m Boys u/19 ',
                     'Bristol House', '2009', 41.13);
end;
    }
//*****************************************************************************************
//Question 2.2.1
//*****************************************************************************************

procedure TfrmQuestion2.btnDisplayCurrentHolderClick(Sender: TObject);
begin
    //Question 2.2.1
  redOutput.Clear;
  redOutput.lines.add(Boys19Relay.ToString);
  btnValidateTime.Enabled := true;
end;
    }
//*****************************************************************************************
//Question 2.2.2
//*****************************************************************************************
procedure TfrmQuestion2.btnValidateTimeClick(Sender: TObject);
var
  sTeam : String;
  rNewTime : Real;
  btnCheckRecord : TButton;
  iCharCounter : integer;
begin
  //Question 2.2.2
  if (edtTime.Text = '') then  //check edtTime is not empty
  begin
      MessageDlg('The Time box is empty.',mtError, [mbOk], 0);

      Exit;
  end
  else
  begin
   for iCharCounter := 1 to length(edtTime.Text) do
    begin
     if NOT(edtTime.Text[iCharCounter] IN ['0'..'9','.']) then
      begin
```

```
        MessageDlg('The value entered invalid.',mtError, [mbOk], 0);
        Exit;
      end;//if
    end;//for
  end;//else

  sTeam := edtTeam.Text;
  rNewTime := StrToFloat(edtTime.Text);
  //Question 2.2.2
  btnCheckRecord  := TButton.Create(grpQ22);
  btnCheckRecord.Parent := grpQ22;
  btnCheckRecord.Left :=  72;
  btnCheckRecord.Top := 158;
  btnCheckRecord.Height := 55;
  btnCheckRecord.Width := 235;
  btnCheckRecord.Caption := 'Check Record';
  btnCheckRecord.OnClick := btnCheckStatus;
end;

procedure TfrmQuestion2.btnCheckStatus(Sender: TObject);
begin
  Boys19Relay.checkForRecord(edtTeam.Text,
                             StrToFloat(edtTime.Text));
  redOutput.Clear;
  redOutput.lines.Add(Boys19Relay.ToString);
  btnRecordStatus.Enabled := true;
end;
    }
//*********************************************************************************************
//Question 2.2.3
//*********************************************************************************************

procedure TfrmQuestion2.btnRecordStatusClick(Sender: TObject);
var
 sOutputString : String;
begin
  //Question 2.2.3
 if (Boys19Relay.getYear = '2013' ) OR  (Boys19Relay.getYear = '2014')
then
     sOutputString := #13 + 'Recent record'
 else
    sOutputString := #13 + 'Old record';
  redOutput.lines.Add(sOutputString);
end;

procedure TfrmQuestion2.btnCloseClick(Sender: TObject);
begin
  Close;
end;
end.
```

ANNEXURE F: SOLUTION FOR QUESTION 3: DELPHI

```
unit Question3Unit;

interface

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

type
  TfrmQuestion3 = class(TForm)
    grpReports: TGroupBox;
    grpQ31: TGroupBox;
    grpQ32: TGroupBox;
    grpQ33: TGroupBox;
    memBackground: TMemo;
    grpSchReport: TGroupBox;
    GroupBoxSelectSchool: TGroupBox;
    btnDisplayReport: TButton;
    cmbSchool: TComboBox;
    btnAverageResults: TButton;
    btnSwapPoints: TButton;
    redOutput: TRichEdit;
    lblHeading: TLabel;
    btnClose: TButton;
    procedure FormCreate(Sender: TObject);
    procedure btnDisplayReportClick(Sender: TObject);
    procedure cmbSchoolChange(Sender: TObject);
    procedure btnAverageResultsClick(Sender: TObject);
    procedure btnSwapPointsClick(Sender: TObject);
    procedure btnCloseClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }

  end;

var
  frmQuestion3: TfrmQuestion3;

implementation

{$R *.dfm}

//*****************************************************************************************
//Given code:
//*****************************************************************************************
var
    arrSchoolNames : array[1..8] of String =
           ('Bedworthpark High School', 'Bristol House',
           'Broadlands Technical High', 'Griffiths House',
           'Fenham College', 'Edenburgh High School',
           'Rethanda College', 'Sheffield High School');
```

```
    arrSchoolAbrv  : array[1..8] of String = ('BPK', 'BSL', 'BRT',
                          'GFH', 'FNH', 'EDB', 'RTN', 'SFD');
    arrResults : array[1..8,1..4] of integer =
                ((365,458,214,0),(255,125,128,0),(489,499,478,0),
                (211,212,256,0),(356,345,387,0),(479,508,479,0),
                (259,245,287,0),(302,315,354,0));

    arrBoys : array[1..8] of integer;
    arrGirls : array[1..8] of integer;
    tempResults : array[1..8,1..4] of integer =
                ((0,0,0,0),(0,0,0,0),(0,0,0,0),(0,0,0,0),
                (0,0,0,0),(0,0,0,0),(0,0,0,0),(0,0,0,0));

procedure TfrmQuestion3.FormCreate(Sender: TObject);
var
  iCounter : integer;
begin
   //populate school names combobox
  for iCounter := 1 to 8 do
  begin
      cmbSchool.Items.Add(arrSchoolAbrv[iCounter]);
  end;//for
end;

//*****************************************************************************************
//Question 3.1
//*****************************************************************************************
procedure TfrmQuestion3.btnDisplayReportClick(Sender: TObject);
var
  K, iPlace: integer;
  TFile : TextFile;
  sLine, sSchool, sSchoolAbrv, sSAbrv : String;
  sGender : String;
const
   sFileName = 'Results.txt';
begin
redOutput.Lines.Clear;
redOutput.Paragraph.TabCount := 4;
redOutput.Paragraph.Tab[0]   := 100;
redOutput.Paragraph.Tab[1]   := 200;
redOutput.Paragraph.Tab[2]   := 300;
redOutput.Paragraph.Tab[3]   := 400;

sSchoolAbrv := cmbSchool.Text;
sSchool := arrSchoolNames[cmbSchool.ItemIndex + 1];
redOutput.Lines.Add('School: ' + sSchool +  '  (' + sSchoolAbrv + ')');
redOutput.Lines.Add('Place' + #9 + 'Boys' + #9 + 'Girls' +  #9 +
'Total');

  for K := 1 to 8 do
   begin
     arrBoys[K] := 0;
     arrGirls[K] := 0;
   end;

  try
  if NOT FileExists(sFileName)then
```

```
    begin
       MessageDlg('The file does not exists.', mtWarning, [mbOk],0);
       Exit;
    end;
  AssignFile(TFile, sFileName);
  Reset(TFile);
  While NOT EOF (TFile) do
    begin
      Readln(TFile, sLine);   //71;Javelin-BSL#Boys u/15;7
      sSAbrv := copy(sLine,pos('#',sLine) - 3, 3);
      sGender := copy(sLine,pos('-',sLine) + 1,1) ;
      iPlace := StrToInt(sLine[length(sLine)]);    // last character
      if (sSAbrv = sSchoolAbrv) then
       if (sGender = 'B')  then
           inc(arrBoys[iPlace])
        else
           inc(arrGirls[iPlace]);
    end; //e while
  finally
   CloseFile(TFile);
  end;
   for K := 1 to 8 do
     redOutput.Lines.Add(IntToStr(K) + #9 + IntToStr(arrBoys[K]) + #9 +
IntToStr(arrGirls[K]) +  #9 + IntToStr(arrBoys[K] + arrGirls[K]));
end;

procedure TfrmQuestion3.cmbSchoolChange(Sender: TObject);
begin
  btnDisplayReport.Enabled := true;
  redOutput.Lines.Clear;
  redOutput.Lines.Add('Place' + #9 + 'Boys' + #9 + 'Girls' +  #9 +
'Total');
end;

//*****************************************************************************************
//Question 3.2
//*****************************************************************************************
procedure TfrmQuestion3.btnAverageResultsClick(Sender: TObject);
var
  iCol, iRow, iTotal : integer;
  iAverage,iHighestAverage : integer;
  sLine : String;
begin
  redOutput.Clear;
  redOutput.Paragraph.TabCount := 5;
  redOutput.Paragraph.Tab[0]   := 80;
  redOutput.Paragraph.Tab[1]   := 180;
  redOutput.Paragraph.Tab[2]   := 280;
  redOutput.Paragraph.Tab[3]   := 380;
  redOutput.Paragraph.Tab[4]   := 480;
  redOutput.Lines.Add('Results of schools over three years');
  iHighestAverage := 0;
  for iCol := 1 to 8 do
     begin
      iTotal := 0;
      for iRow := 1 to 3 do
        iTotal := iTotal + arrResults[iCol][iRow];
```

```
      iAverage :=  Trunc(iTotal / 3);
      arrResults[iCol][4] := iAverage;
      if iHighestAverage < iAverage then
        iHighestAverage := iAverage;
     end;   // for iCol

    redOutput.Lines.Add(' ' + #9 + '2012' + #9 + '2013' +  #9 + '2014'+
#9 + 'Average points');
    for iCol := 1 to 8 do
      begin
        if arrResults[iCol][4] = iHighestAverage then
            arrSchoolAbrv[iCol]  := arrSchoolAbrv[iCol] + '*';
        sLine := arrSchoolAbrv[iCol];
        for iRow := 1 to 4 do
          sLine := sLine + #9 + IntToStr(arrResults[iCol][iRow]);
      redOutput.Lines.Add(sLine);
      end; // for
end;

//*****************************************************************************************
//Question 3.3
//*****************************************************************************************
procedure TfrmQuestion3.btnSwapPointsClick(Sender: TObject);
var
  K :integer;
begin
  for K := 1 to 4 do
     tempResults[1][K] := arrResults[1][K];
  arrResults[1] := arrResults[2];
  for K := 1 to 4 do
     arrResults[2][K] := tempResults[1][K];
  btnAverageResults.Click;
end;

procedure TfrmQuestion3.btnCloseClick(Sender: TObject);
begin
  Close;
end;

end.
```

ANNEXURE G: SOLUTION FOR QUESTION 1: JAVA

```
//*************************************************************************************************
//Global variables
//*************************************************************************************************
String id;

//*************************************************************************************************
//Question 1.1
//*************************************************************************************************
private void btnQuest1_1ActionPerformed(java.awt.event.ActionEvent evt)
{
        // Question 1.1.
        lblHeading.setText("PC Athletics Championships");
    }
//*************************************************************************************************
//Question 1.2
//*************************************************************************************************
private void btnQuest1_2ActionPerformed(java.awt.event.ActionEvent evt)
{
        // Question 1.2
        String name = txfFullNames.getText();
        String surname = txfSurname.getText();
        id = txfID.getText();
        String initials = "" + name.charAt(0);
        for (int i = 0; i < name.length(); i++) {
            if (name.charAt(i) == ' ') {
                initials += name.charAt(i + 1);
            }
        }
        int genderNum = Integer.parseInt(id.substring(6, 10));
        String gender = "Female";
        if (genderNum >= 5000) {
            gender = "Male";
        }
        String tg = surname + " " + initials + ".";
        txaQ1.setText("Athlete: " + tg.toUpperCase() + "\n\t\t" +
gender);
    }

//*************************************************************************************************
//Question 1.3
//*************************************************************************************************
private void btnQuest1_3ActionPerformed(java.awt.event.ActionEvent evt)
{
        // Question 1.3
        String output = "\nAge group: ";
        int yrOfBirth = 0;
        int year = Integer.parseInt(txfID.getText().substring(0, 2));
        if (year <= 14) {
            yrOfBirth = 2000 + year;
        } else {
            yrOfBirth = 1900 + year;}
        int age = 2014 - yrOfBirth;
        switch (age) {
```

```
            case 14:
            case 15:
                output += "u/15";
                break;
            case 16:
            case 17:
                output += "u/17";
                break;
            case 18:
            case 19:
                output += "u/19";
                break;
            default:
                output += "Not eligible to participate";
        }
        txaQ1.setText(txaQ1.getText() + output);
    }

//*************************************************************************************************
//Question 1.4
//*************************************************************************************************
DecimalFormat df = new DecimalFormat("R 0.00");
        String output;
        double price = Double.parseDouble
(JOptionPane.showInputDialog("Please type unit price: "));
        int quantity = Integer.parseInt
(JOptionPane.showInputDialog("Please type in the quantity: "));
        output = String.format("%-8s%-3s%-8s%-5s%-8s%-10s%-12s%n",
          "No", "", "Price", "", "Total", "Discount", "Amount Due");
        txaQ1.setText(output);
        double discount = 0;
        double totalDiscount = 0;
        for (int cnt = 1; cnt <= quantity; cnt++) {

            double cost = price * cnt;
            if (cnt % 2 == 0) {
                discount = price * cnt * 20 / 100;
            }

            output = String.format("%-8d%-3s%-8.2f%-5s%-8.2f%-10.2f%-
10.2f%n", cnt, "X", price, "= R", (price * cnt), discount, ((price *
cnt) - discount));
            txaQ1.append(output);
        }
        double amountTendered = Double.parseDouble
(JOptionPane.showInputDialog("Enter the amount tendered"));
        double change = amountTendered - ((price * quantity) -
discount);

        int rands, coins50, coins20, coins10;
        rands = (int) (change);
        double centsD = Math.round((change - rands) * 100);
        int cents = (int) (centsD);
        coins50 = cents / 50;
        cents %= 50;
        coins20 = cents / 20;
        cents %= 20;
        coins10 = cents / 10;
        cents %= 10;
```

```
        if(cents > 0)
        {
            coins10++;
        }
        txaQ1.append(String.format("%-15s%-10.2f%n", "Change:",
change));
        if (rands > 0) {
            txaQ1.append(String.format("%-15s%-10s%n", "Rands:",
rands));
        }
        if (coins50 > 0) {
            txaQ1.append(String.format("%-15s%-10s%n", "50c coins:",
coins50));
        }
        if (coins20 > 0) {
            txaQ1.append(String.format("%-15s%-10s%n", "20c coins:",
coins20));
        }
        if (coins10 > 0) {
            txaQ1.append(String.format("%-15s%-10s%n", "10c coins:",
coins10));
        }
```

ANNEXURE H: SOLUTION FOR QUESTION 2: JAVA

```
//*************************************************************************************************
//Question 2.1 RelayEvent object class:
//*************************************************************************************************
package SolutionQ2Package;
import java.util.Calendar;
public class RelayEvent {

    private String event;
    private String team;
    private String year;
    private double recordTime;
    public RelayEvent(String event, String team, String year, double
recordTime) {
        this.event = event;
        this.team = team;
        this.year = year;
        this.recordTime = recordTime;
    }
    public String getEvent() {
        return event;
    }
    public String getTeam() {
        return team;
    }
    public String getYear() {
        return year;
    }
    public double getRecordTime() {
        return recordTime;
    }
    public void checkForRecord(String newTeam, double newTime) {
        Calendar rightNow = Calendar.getInstance();
        int yr = rightNow.getWeekYear();
        if (newTime == recordTime) {
            team = team + ";" + newTeam;
            year = year + ";" + yr;
        }
        if (newTime < recordTime) {
            recordTime = newTime;
            team = newTeam;
            year = "" + yr;
        }
    }

    public String toString() {
        return "Current record for " + event + ":\n\nTeam: " + team +
                "\nYear: " + year + "\nTime: " + recordTime + "
seconds\n";}
}
```

```
//*****************************************************************************************
//Question 2.2 Driver class:
//*****************************************************************************************
private javax.swing.JButton btnCheckRecord;
    String tName = "";
    double nTime = 0;
    RelayEvent boys19Relay;

    public Question2Memo() {
        initComponents();
        setLocationRelativeTo(this);
        this.setVisible(true);
    }

//*****************************************************************************************
//Question 2.2.1
//*****************************************************************************************
private void btnQ2_1ActionPerformed(java.awt.event.ActionEvent evt) {
        boys19Relay = new RelayEvent("Boys u/19 4x100m relay", "Bristol
House", "2009", 41.13);
        txaQ2.setText(boys19Relay.toString());
    }

//*****************************************************************************************
//Question 2.2.2
//*****************************************************************************************
private void btnQ2_2ActionPerformed(java.awt.event.ActionEvent evt) {
       //Question 2.2:
        nTime = 0;
        tName = txfTeam.getText();
        try {
            nTime = Double.parseDouble(txfTime.getText());
             createRecordButton();
             txaQ2.setText("");
        } catch (NumberFormatException f) {
            txaQ2.setText("Time is not in a correct format, please
retype");
            txfTime.setText("");
        }
    }

//*****************************************************************************************
//Question 2.2.3
//*****************************************************************************************
private void btnQ2_3ActionPerformed(java.awt.event.ActionEvent evt) {
          String output = "\nRecent record ";
            if (boys19Relay.getYear().indexOf("2013") ==-1 ||
            boys19Relay.getYear().indexOf("2014") ==-1)
            {
             output ="\nOld record ";
             txaQ2.append(output);
    }
}

private void createRecordButton() {
        pnlQ2_2.setOpaque(true);
        btnCheckRecord = new JButton();           //makes the button
```

```
        pnlQ2_2.add(btnCheckRecord);        // adds the button to the panel
        btnCheckRecord.setText("Check Record");  // sets text on the button
        btnCheckRecord.setBounds(35, 235, 205, 55);
                                 //Xvalue,Yvalue,Width,Height
        btnCheckRecord.setVisible(true);         //shows the button
        btnCheckRecord.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                txaQ2.setText("");
                boys19Relay.checkForRecord(tName, nTime);
                txaQ2.setText(boys19Relay.toString());
            }
        });
    }
```

ANNEXURE I: SOLUTION FOR QUESTION 3: JAVA

```
//*****************************************************************************************
//Given code
//*****************************************************************************************
public class Question3Memo extends javax.swing.JFrame {

    int[] arrGirls = new int[8];
    int[] arrBoys = new int[8];
    String[] arrSchoolNames = {"Bedworthpark High School", "Bristol
House", "Broadlands Technical High", "Griffiths House", "Fenham
College", "Edenburgh High School", "Rethanda College", "Sheffield High
School"};
    String[] arrSchoolAbrv = {"BPK", "BSL", "BRT", "GFH", "FNH", "EDB",
                              "RTN", "SFD"};
    int[][] arrSchoolResults = {{365, 458, 214, 0}, {255, 125, 128, 0},
                                {489, 499, 478, 0}, {211, 212, 256, 0},
                                {356, 345, 387, 0}, {479, 508, 479, 0},
                                {259, 245, 287, 0}, {302, 315, 354, 0}};

    public void fillComboBox() {
        for (int cnt = 0; cnt < 8; cnt++) {
            cbxSchool.addItem("" + arrSchoolAbrv[cnt]);}
    }

    public Question3Memo() {
        initComponents();
        setLocationRelativeTo(this);
        this.setVisible(true);
        fillComboBox();
    }

//*****************************************************************************************
//Question 3.1
//*****************************************************************************************
private void btnQ3_1ActionPerformed(java.awt.event.ActionEvent evt) {
        int posnSchool = cbxSchool.getSelectedIndex();
        String schoolAbr = arrSchoolAbrv[posnSchool];

        try {
          BufferedReader bf = new BufferedReader(new
FileReader("Results.txt"));

            for (int cnt = 0; cnt < 8; tel++) {
                arrBoys[cnt] = 0;
                arrGirls[cnt] = 0;
            }
            String line = bf.readLine();
            while (line != null) {
                line = line.replace(";", "#");
                String[] temp = line.split("#");
                if (temp[1].equals(schoolAbr)) {
                    if (temp[2].indexOf("Boys") >= 0) {
                        int position = Integer.parseInt(temp[3]);
                        arrBoys[position - 1]++;
```

```
                    }
                    if (temp[2].indexOf("Girls") >= 0) {
                        int position = Integer.parseInt(temp[3]);
                        arrGirls[position - 1]++;
                    }
                }
                line = bf.readLine();
            }
            String output = String.format("%-12s%-12s%-12s%-12s%n",
                            "Place", "Boys", "Girls", "Total");
            txaOutput.setText("School: " + arrSchoolNames[posnSchool] +
" (" + schoolAbr + ")\n" + output);
            for (int tel = 0; tel < 8; tel++) {
                output = String.format("%-12d%-12d%-12d%-12d%n", (tel +
1),
                arrBoys[tel], arrGirls[tel], (arrBoys[tel] +
arrGirls[tel]));
                txaOutput.append(output);
            }
        } catch (FileNotFoundException e) {
            System.out.println(e);
        } catch (Exception f) {
            System.out.println(f);
        }
    }

//*****************************************************************************************
//Question 3.2
//*****************************************************************************************
private void btnQ3_2ActionPerformed(java.awt.event.ActionEvent evt) {

        txaOutput.setText("Average results of schools over the past
three years\n");
        for (int i = 0; i < 8; i++) {
            arrSchoolResults[i][3] = 0;
        }
        int scoreHigh = 0;
        String output = String.format("%-20s%-12s%-12s%-12s%-12s%n", "",
"2012", "2013", "2014", "Average points");
        txaOutput.append(output);

        for (int schCnt = 0; schCnt < 8; schCnt++) {
            for (int yrCnt = 0; yrCnt < 3; yrCnt++) {
                arrSchoolResults[schCnt][3] =
arrSchoolResults[schCnt][3] +
                arrSchoolResults[schCnt][yrCnt];
            }

          arrSchoolResults[schCnt][3] = arrSchoolResults[schCnt][3] / 3;
          if (scoreHigh < arrSchoolResults[schCnt][3]) {
              scoreHigh = arrSchoolResults[schCnt][3];
            }
        }
        for (int schCnt = 0; schCnt < 8; schCnt++) {
            String school = arrSchoolAbrv[schCnt];
            if (scoreHigh == arrSchoolResults[schCnt][3]) {
                school = arrSchoolAbrv[schCnt] + "*";
            }
```

```
            output = String.format("%-20s%-12d%-12d%-12d%-12d%n", school,
                     arrSchoolResults[schCnt][0],
arrSchoolResults[schCnt][1],
                     arrSchoolResults[schCnt][2],
arrSchoolResults[schCnt][3]);
             txaOutput.append(output);
        }
    }

//*****************************************************************************************
//Question 3.3
//*****************************************************************************************
private void btnQ3_3ActionPerformed(java.awt.event.ActionEvent evt) {
        int[] tempResults   = arrSchoolResults[0];
        arrSchoolResults[0] = arrSchoolResults[1];
        arrSchoolResults[1] = tempResults;
        btnQ3_2.doClick();
    }
```

TOTAL: 150