Province of the

EASTERN CAPE

EDUCATION

NATIONAL SENIOR CERTIFICATE

GRADE 12

SEPTEMBER 2012

INFORMATION TECHNOLOGY P1 MEMORANDUM

MARKS: 120

This memorandum consists of 11 pages.

SECTION A:

QUESTION 1 (30 marks)

(LO 4 AS 11, 12)

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Select *✓ from EducatorsTb ✓order by Surname✓';
  qrySchool.Active := true;
  SetGridColumnWidths(DBGrid1);
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Select Title, Surname, TeacherCode✓ from EducatorsTb✓ where
(subjects = "ENG"✓ or subjects = "AFR"✓ or subjects = "XHO"✓) ';
  qrySchool.Active := true;                                    OR where (subjects IN
[“ENG”,”AFR”,”XHO”)
  SetGridColumnWidths(DBGrid1);
end;

procedure TForm1.Button3Click(Sender: TObject);
var
  ssubject : string;
begin
  ssubject := inputbox('Enter Subject','',''); ✓
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Select count(*)✓as [Potential Subject Numbers] ✓ from LearnersTb✓
where (S5 = "'+ssubject+'"')'; ✓
  qrySchool.Active := true;
  SetGridColumnWidths(DBGrid1);
end;

procedure TForm1.Button5Click(Sender: TObject);
begin
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Select Name, Surname✓ from LearnersTb ✓where Teachers like
"%AM%"'; ✓
  qrySchool.Active := true;
  SetGridColumnWidths(DBGrid1);
end;

procedure TForm1.Button7Click(Sender: TObject);
begin
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Update LearnersTb✓ Set S5 = "LSC"✓ where S5 = "ART"'; ✓
  qrySchool.Active := true;
  qrySchool.SQL.Text := 'Select *✓ from LearnersTb'; ✓
  SetGridColumnWidths(DBGrid1);
end;
```

```
procedure TForm1.Button8Click(Sender: TObject);
begin
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Select class, count(class) ✓ AS [Class Sizes] ✓ from LearnersTb✓ group
by Class✓';
  qrySchool.Active := true;
  SetGridColumnWidths(DBGrid1);
end;

procedure TForm1.Button9Click(Sender: TObject);
begin
  qrySchool.active := false;
  qrySchool.SQL.Text := 'Select *✓ from LearnersTb ✓where (S7 = "CONS" ✓or S7 = "TOUR"✓)
and (class = "G"✓)';
  qrySchool.Active := true;
  SetGridColumnWidths(DBGrid1);
end;
```

QUESTION 2 (55 marks)

2.1	**(LO4 AS4)**		
	2.1.1	Define a class Private Declaring fname, feng, fmaths, faverage	(1) (1) (2)
	2.1.2	Constructor heading Assigning values to fields Initialising faverage	(1) (3) (1)
	2.1.3	Procedure CalcAverage heading calculation	(1) (2)
	2.1.4	Function Qualify – Boolean If average >= 60 Qualify true else qualify false	(1) (1) (2)
	2.1.5	Function tostring heading Putting fields together #9 for columns	(1) (1) (1)
	2.1.6	GetName heading fname assigned to result/function	(1) (1)
	2.1.7	GetAverage heading faverage assigned to result/function	(1) (1)
2.2	2.2.1	Initialise counter Check if file exists Assignfile Reset While not eof() do Readln Increase counter Get name Get eng mark Get maths mark Assign to arrlearner	(1) (1) (1) (1) (1) (1) (1) (1) (1) (1) (2)
	2.2.2	Loop Call CalcAverage Call getname and getaverage → display	(1) (1) (2)
	2.2.3	Loop Call CalcAverage Call qualify If qualify = true display tostring	(1) (1) (1) (1)
	2.2.4	Inputbox Set flag variable to false Loop (counter less than number of learners and flag variable false) If getname = input from inputbox Then flag variable = true Else flag variable = false Increase counter If flag variable = true then display that name was found	(1) (1) (2) (2) (1) (1) (1) (2)
			[32]

POSSIBLE SOLUTION

```
unit Subjects_u;

interface

uses
  SysUtils;

type
  TLearner = class
   private
    fname : string;
    feng : integer;
    fmaths : integer;
    faverage : real;
   public
    constructor create (sname : string; ieng, imaths : integer);
    procedure CalcAverage;
    function tostring : string;
    function getname : string;
    function getaverage : real;
    function qualify : boolean;
  end;
implementation

constructor TLearner.create (sname : string; ieng, imaths : integer);
begin
   fname := sname;
   feng := ieng;
   fmaths := imaths;
   faverage := 0;
end;

procedure TLearner.CalcAverage;
begin
  faverage := (feng + fmaths)/2;
end;

function Tlearner.tostring : string;
begin
  result := fname + #9 + inttostr(feng) + #9 + inttostr(fmaths) + #9 +
floattostrf(faverage,fffixed,3,1);
end;

function TLearner.getname : string;
begin
  result := fname;
end;

function TLearner.getaverage : real;
begin
  result := faverage;
end;

function TLearner.qualify : boolean;
begin
   if faverage >= 60 then
```

```
    qualify := true
  else
    qualify := false;
end;

end.
```

unit Question2_u;

```
interface

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

type
  TForm1 = class(TForm)
    MainMenu1: TMainMenu;
    RichEdit1: TRichEdit;
    Options1: TMenuItem;
    Average1: TMenuItem;
    ITPhysicalScienceMaths1: TMenuItem;
    SearchforaLearner1: TMenuItem;
    Exit1: TMenuItem;
    procedure FormActivate(Sender: TObject);
    procedure Average1Click(Sender: TObject);
    procedure ITPhysicalScienceMaths1Click(Sender: TObject);
    procedure SearchforaLearner1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
  icount : integer;
  arrLearners : array[1..50] of TLearner;
implementation

{$R *.dfm}

procedure TForm1.FormActivate(Sender: TObject);
var
  myfile : textfile;
  soneline : string;
  ipos, ieng, imaths : integer;
  sname : string;
begin
  if fileexists('Learner.txt') <> true then
    begin
      ShowMessage('File does not exist');
      Exit;
    end;
  Assignfile(myfile, 'Learner.txt');
  Reset(myfile);
  icount := 0;
  while not eof(myfile) do
    begin
      readln(myfile,soneline);
      inc(icount);
      ipos := pos(',',soneline);
      sname := copy(soneline, 1, ipos - 1);
      delete(soneline, 1, ipos);
      ipos := pos(',',soneline);
      ieng := strtoint(copy(soneline, 1, ipos - 1));
      delete(soneline, 1, ipos);
      imaths := strtoint(soneline);
      arrLearners[icount] := TLearner.create(sname, ieng, imaths);
    end;
  closefile(myfile);
end;

procedure TForm1.Average1Click(Sender: TObject);
var
  k : integer;
begin
  richedit1.lines.add('Name' + #9 + #9 + 'Average');
  for k := 1 to icount do
   begin
    arrLearners[k].CalcAverage;
    richedit1.Lines.add(arrLearners[k].getname + #9 +
floattostrf(arrLearners[k].getaverage,fffixed,6,0));
   end;

procedure TForm1.ITPhysicalScienceMaths1Click(Sender: TObject);
var
  k : integer;
begin
  richedit1.lines.add('Name' + #9 +#9+ 'English' + #9 + 'Maths' + #9 + 'Average');
  for k := 1 to icount do
    begin
      arrLearners[k].CalcAverage;
      if arrlearners[k].qualify then
        richedit1.Lines.Add(arrlearners[k].tostring);
    end;

procedure TForm1.SearchforaLearner1Click(Sender: TObject);
var
  sinput : string;
  k : integer;
  bfound : boolean;
begin
  sinput := inputbox('Enter Name','','');
  bfound := false;
  k := 1;
  while (k <= icount) and (bfound = false) do
    begin
      if arrLearners[k].getname = sinput then
        bfound := true
       else
        bfound := false;
      inc(k);
    end;
    if bfound = true then
      ShowMessage(sinput + ' found.');
    if bfound = false then
      ShowMessage(sinput + ' not found.');
end;
```

QUESTION 3 (35 marks)

3.1	For loop – columns For loop – rows Assign random values (100 – 600) to stringgrid Assign column headings – using a for loop (2), alternative method (1) Assign row headings – using a for loop (2), alternative method (1)	(1) (1) (2) (2) (2)
3.2	Procedure Calculate Totals Heading for row in stringgrid For loop – column Initialise total variable For loop – row Calculate total of all values in each column	(1) (1) (1) (1) (1) (1)
3.3	Procedure CalculateAvgWeek Heading for row in stringgrid For loop Initialise total variable Add all values in columns (for each class) Divide totals by 7 Display the average formatted to 2 decimals in the stringgrid	(1) (1) (1) (1) (2) (2) (1)
3.4	CalculateTotals call statement CalculateAvgWeek call statement	(1) (1)
3.5	Initialise high variable to 0 For loop – row Initialise sum variable For loop – column Add values for each class If sum is higher than high Assign sum’s value to high Assign the row value to a variable Display message of winning class	(1) (1) (1) (1) (1) (1) (1) (1) (1)
		[35]

POSSIBLE SOLUTION

```
unit question3_u;

interface

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

type
  TForm1 = class(TForm)
    StringGrid1: TStringGrid;
    Button1: TButton;
    Button2: TButton;
    procedure FormCreate(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
  publi
    procedure CalculateTotals;
    procedure CalculateAvgWeek;
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.FormCreate(Sender: TObject);
var
  irow, icol : integer;
begin
  randomize;
  for icol := 1 to 4 do
    for irow := 1 to 7 do
      stringgrid1.Cells[icol,irow] := inttostr(random(500)+101);

  for icol := 1 to 4 do
    stringgrid1.Cells[icol,0] := 'Week '+inttostr(icol);

  for irow := 1 to 7 do
    stringgrid1.cells[0,irow] := 'Class '+inttostr(irow);

end;

procedure TForm1.CalculateTotals;
var
  icol,irow, itotal : integer;
begin
  stringgrid1.cells[0,8] := 'Totals';
  for icol := 1 to 4 do
    begin
     itotal := 0;
```

```
      for irow := 1 to 7 do
        begin
          itotal := itotal + strtoint(stringgrid1.cells[icol,irow]);
        end;
      stringgrid1.Cells[icol,irow] := inttostr(itotal)
    end;
end;

procedure TForm1.CalculateAvgWeek;
var
  irow, icol : integer;
  rtotal, raverage : real;
begin
  stringgrid1.cells[0,9] := 'Average';
  for icol := 1 to 4 do
    begin
       rtotal := rtotal + strtofloat(stringgrid1.cells[icol,8]);
       raverage := rtotal/7;
       stringgrid1.cells[icol,9] := floattostrf(raverage,fffixed,6,2);
       rtotal := 0;
    end;

end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  CalculateTotals;
  CalculateAvgWeek;
end;

procedure TForm1.Button2Click(Sender: TObject);
var
  irow, ihigh, iclass, icol, isum,k,l,itemp : integer;
  arrclass : array[1..7] of integer;
begin
  ihigh := 0;
  for irow := 1 to 7 do
    begin
      isum := 0;
      for icol := 1 to 4 do
        begin
        isum := isum + strtoint(stringgrid1.cells[icol,irow]);
        end;
        if isum > ihigh then
         begin
          ihigh := isum;
          iclass := irow;
         end;
    end;
  ShowMessage('Winning class: '+inttostr(iclass));
end;

end.
```