



Province of the
EASTERN CAPE
EDUCATION

**NASIONALE
SENIOR SERTIFIKAAT**

GRAAD 11

NOVEMBER 2015

**INLIGTINGSTEGNOLOGIE V1
MEMORANDUM**

PUNTE: 150

Hierdie memorandum bestaan uit 11 bladsye.

VOORBEELD DELPHI OPLOSSING VRAAG 1

```

unit Question1_u;
//Enter your Name and Surname Here

interface

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

type
  TfrmQuestion1 = class(TForm)
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    TabSheet3: TTabSheet;
    btnLanes: TButton;
    bbClose: TBitBtn;
    memLanes: TMemo;
    btnAdd: TButton;
    btnDisplay: TButton;
    redDisplay: TRichEdit;
    lblID: TLabel;
    edtID: TEdit;
    btnCheck: TButton;
    memID: TMemo;
    procedure btnCheckClick(Sender: TObject);
    procedure btnAddClick(Sender: TObject);
    procedure btnLanesClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure btnDisplayClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuestion1: TfrmQuestion1;
  icount, itotal : integer;
  rlongest, rtotal : real;
implementation

{$R *.dfm}

///////// Question 1.1
procedure TfrmQuestion1.btnLanesClick(Sender: TObject);
var
  ihouse1, ihouse2, ihouse3 : integer;
begin
  randomize;
  repeat✓
    ihouse1 := random(3)+1; ✓
    ihouse2 := random(3)+1; ✓
    ihouse3 := random(3)+1; ✓
  until (ihouse1 <> ihouse2) and✓ (ihouse1 <> ihouse3); ✓

```

```

    memlanes.lines.add('House 1: is in Lane' +inttostr(ihouse1) ✓+' and
Lane '+inttostr(ihouse1+3)); ✓
    memlanes.lines.add('House 2: is in Lane' +inttostr(ihouse2) +' and
Lane '+inttostr(ihouse2+3)); ✓
    memlanes.lines.add('House 3: is in Lane' +inttostr(ihouse3)+' and
Lane '+inttostr(ihouse3+3)); ✓
end;

```

```

    ///Question 1.2.1
procedure TfrmQuestion1.btnAddClick(Sender: TObject);
var
    rdistance : real;
begin
    rdistance := strtofloat(Inputbox('Add Distance',' ','')); ✓
    inc(icount); ✓
    rtotal := rtotal + rdistance; ✓
    if rdistance > rlongest then ✓
        rlongest := rdistance; ✓
end;
procedure TfrmQuestion1.FormCreate(Sender: TObject);
begin
    rlongest := 0;
    icount := 0; ✓
    rtotal := 0;
    itotal := 0; ✓
end;

```

```

/////1.2.2 (6)
procedure TfrmQuestion1.btnDisplayClick(Sender: TObject);
var
    raverage : real;
begin
    raverage := rtotal/icount; ✓
    reddisplay.lines.add('Total: '+inttostr(icount)); ✓
    reddisplay.lines.add('Average: '+floattostrf ✓
(raverage,ffixed,8,2)); ✓
    reddisplay.lines.add ✓ ('Longest: '+floattostr(rlongest)); ✓
end;

```

```

/////Question 1.2.3 (24)
procedure TfrmQuestion1.btnCheckClick(Sender: TObject);
var
    sid : string;
    ilen,k : integer;
    bvalid : boolean;
begin
    sid := edtid.text; ✓
    ilen := length(sid); ✓
    bvalid := true; ✓
    for k := 1 to ilen do ✓
        begin
            if not(sid[k] in ['0'..'9']) then ✓
                bvalid := false; ✓
        end;
    end;

```

```
if (bvalid = false) ✓and (ilen <> 13) then✓  
    Showmessage('Enter numbers only and not 13 digits') ✓  
else✓  
if ilen <> 13 then✓  
    Showmessage('too short') ✓  
else✓  
    if bvalid = false then✓  
        Showmessage('Enter numbers only') ✓  
    else✓  
        begin  
            Showmessage('Valid ID'); ✓  
            for k := 1 to ilen do✓  
                begin  
                    itotal := itotal + strtoint(sid[k]); ✓  
                end;  
            if itotal mod 2 = 0 then✓  
                memid.lines.add('Even: You can have another throw')  
            else✓  
                memid.Lines.add('Odd: Sorry, no extra throw'); ✓  
            end;  
        end;  
    end;  
end;  
  
end.
```

[47]

VOORBEELD DELPHI OPLOSSING VRAAG 2

```

unit Question2_U; //Enter your name and surname Here
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Buttons, StdCtrls, ExtCtrls, Math, ComCtrls;

type
  TfrmQ2 = class(TForm)
    pnlOutPut: TPanel;
    pnlChoices: TPanel;
    btnRead: TButton;
    bbnClose: TBitBtn;
    btnListOfSponsors: TButton;
    redOutput: TRichEdit;
    procedure btnReadClick(Sender: TObject);
    procedure btnListOfSponsorsClick(Sender: TObject);
  private
    { Private declarations }

  public
    { Public declarations }
  end;
///2.1 (3)
var
  frmQ2: TfrmQ2;
  arrathletes : array[1..20] ✓ of string; ✓
  icount : integer; ✓
implementation

{$R *.dfm}
///2.2 (15)
function validcode(scode : string✓) : boolean; ✓
var
  ilen,k : integer;
begin
  result := true; ✓
  ilen := length(scode); ✓
  for k := 1 to 3 do ✓
    if not (scode[k] in ['A'..'Z']) then ✓
      result := false; ✓
  for k := 4 to 8 do ✓
    if not (scode[k] in ['0'..'9']) then ✓
      result := false; ✓
  for k := 9 to 11 do ✓
    if not (scode[k] in ['A'..'Z']) then ✓
      result := false; ✓
  if ilen <> 11 then ✓
    result := false; ✓

end;
///2.3 (19)
procedure TfrmQ2.btnReadClick(Sender: TObject);
var
  myfile : textfile;
  soneline : string;

```

```

    iinvalid : integer;
begin
    icount := 0; ✓
    iinvalid := 0; ✓
    if fileexists('field.txt') <> true then ✓
        begin
            Showmessage('no'); ✓
            Exit; ✓
        end;
    Assignfile(myfile, 'field.txt'); ✓
    Reset(myfile); ✓
    while not eof(myfile) do ✓
        begin
            readln(myfile, soneline); ✓
            if validcode(soneline) then ✓
                begin
                    inc(icount) ✓;
                    arrathletes[icount] := soneline; ✓
                end
            else
                begin
                    redoutput.lines.add('Invalid: '+soneline); ✓
                    inc(iinvalid); ✓
                end;
            end;
        redoutput.lines.add(''); ✓
        redoutput.lines.add('Valid Codes: '+inttostr(icount)); ✓
        redoutput.Lines.add('Invalid Codes: '+inttostr(iinvalid)); ✓

        closefile(myfile); ✓
    end;

///2.4 (11)
procedure TfrmQ2.btnListOfSponsorsClick(Sender: TObject);
var
    k : integer;
    ssponsor, slist : string;
begin
    for k := 1 to icount do ✓
        begin
            ssponsor := copy(arrathletes[k], 9, 3); ✓
            if pos(ssponsor, slist) = 0 then ✓
                slist := slist + ssponsor; ✓
            end;
        for k := 1 to length(slist) ✓ div 3 do ✓
            redoutput.lines.Add(inttostr(k) ✓ + copy(slist, k*3-2, 3)); ✓
        redoutput.lines.Add(''); ✓
        redoutput.lines.Add('Total: '+inttostr ✓ (length(slist) div 3)); ✓

    end;

end.

```

VOORBEELD DELPHI OPLOSSING VRAAG 3

```
unit Question3_u;

interface

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

type

  TfrmQ3 = class(TForm)
    pnlOutPut: TPanel;
    lbxID: TListBox;
    bmbClose: TBitBtn;
    btnFindFastest: TButton;
    redOutput: TRichEdit;
    btnGender: TButton;
    procedure FormCreate(Sender: TObject);
    procedure SortResults;
    procedure btnFindFastestClick(Sender: TObject);
    procedure btnGenderClick(Sender: TObject);

  private
    { Private declarations }

    NewPanel : TPanel;

  public
    { Public declarations }

  end;

var
  frmQ3: TfrmQ3;
  arrResults : Array[1..100] of string;
  iNumRes : Integer;
implementation

{$R *.dfm}
//=====
=====
procedure TfrmQ3.FormCreate(Sender: TObject);
var iCounter: Integer;
begin
  // Do not alter or add code in this procedure
  // read data from hidden list box into the array
  iNumRes := 0;
  For iCounter := 0 to lbxID.Items.Count -1 do
    arrResults[iCounter+1] := lbxID.Items[iCounter];
  iNumRes := lbxID.Items.Count;
end;
```

```
////3.1 (8)
procedure TfrmQ3.SortResults; ✓
var
  k, l : integer;
  stemp : string;
begin
  for k := 1 to inumres - 1 do ✓
    for l := k + 1 to inumres do ✓
      if arrresults[k] > arrresults[l] then ✓
        begin
          stemp := arrresults[k]; ✓
          arrresults[k] := arrresults[l]; ✓
          arrresults[l] := stemp; ✓
        end;
      end;
end;

////3.2 (5)
procedure TfrmQ3.btnFindFastestClick(Sender: TObject);
begin
  SortResults; ✓
  redoutput.Lines.add('Code ✓ '+arrresults[1]); ✓
  redoutput.Lines.add('Time: ✓' + floattostr(strtoint(copy ✓
(arrresults[1],1,5))/1000)) ✓;
end;

////(3.3)-(9)
procedure TfrmQ3.btnGenderClick(Sender: TObject);
var
  k, imale, ifemale, igender : integer;
begin
  for k := 1 to inumres do ✓
    begin
      igender := strtoint(copy(arrresults[k],8,1)); ✓
      if igender >= 5 then ✓
        inc(imale) ✓
      else ✓
        inc(ifemale); ✓
      end;
    end;
  redoutput.Lines.add('Number of Males: ✓ '+inttostr(imale)) ✓;
  redoutput.Lines.add('Number of Females: '+inttostr(ifemale)) ✓;
end;

end.
```


VOORBEELD DELPHI OPLOSSING VRAAG 4

```

unit Question4_u;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, Grids, DBGrids, dmResultsData_u, (*LEAVE THIS IN*)
  StdCtrls, ExtCtrls, Buttons;

type
  TfrmFour = class(TForm)
    dbgVisitors: TDBGrid;
    btnA: TButton;
    btnC: TButton;
    btnB: TButton;
    btnD: TButton;
    bmbClose: TBitBtn;
    ledName: TLabeledEdit;
    btnE: TButton;
    ledSurname: TLabeledEdit;
    ledAge: TLabeledEdit;
    ledEvent: TLabeledEdit;
    LedPosition: TLabeledEdit;
    ledGender: TLabeledEdit;
    ledHouse: TLabeledEdit;
    ledEarnings: TLabeledEdit;
    Button1: TButton;
    procedure btnAClick(Sender: TObject);
    procedure btnEClick(Sender: TObject);
    procedure btnCClick(Sender: TObject);
    procedure btnBClick(Sender: TObject);
    procedure btnDClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmFour: TfrmFour;

implementation

{$R *.dfm}

procedure TfrmFour.btnAClick(Sender: TObject); (4)
begin
  with dmData.tblWinners do
    begin
      Filter := 'House = ''House1'''; ✓
      Filtered := True; ✓
    end;
end;

```

```

    ShowMessage('Number of Times House 1 appears: ' +
IntToStr(dmData.tblWinners.RecordCount)); ✓
    dmData.tblWinners.Filtered := False; ✓
end;

procedure TfrmFour.btnBClick(Sender: TObject); (6)
var iSum : Integer;
begin
    iSum := 0; ✓
    with dmData do
        begin
            tblWinners.Open; ✓
            tblWinners.First;
            while not tblWinners.Eof do ✓
                begin
                    iSum := iSum + tblWinners['Earning']; ✓
                    tblWinners.Next; ✓
                end;
            end;
            ShowMessage('Total Earnings by athletes : R' + IntToStr(iSum)); ✓
        end;
end;

procedure TfrmFour.btnCClick(Sender: TObject); (3)
begin
    dmData.tblWinners.open; ✓
    dmData.tblWinners.Sort ✓ := 'House DESC, ✓ Event ASC'; ✓
end;

procedure TfrmFour.btnDClick(Sender: TObject); (7)
begin
    with dmData do ✓
        if tblWinners.Locate ✓ ('athSurname',inputbox('Enter Surname to
Search','Surname','Thambo')) ✓, [])
        then
            begin
                ShowMessage('Athlete' + tblWinners['athName']) ✓ +
tblWinners['athSurname'] ✓ + ' has been found'); ✓
            end
            else ShowMessage('No such Surname appears in the database'); ✓
        end;
end;
{Alternate solution to question 4.4
var
    sname:string;
    bfind:boolean;
begin
    sname:=inputbox('','','');
    bfind:=false;
    with dmdata do
        begin
            tblWinners.Open;
            tblWinners.First;
            while no tblWinners.Eof do

```

```

        begin
            if sname = tblwinners['athSurname'] then
                bfind:=true;
                tblwinners.next;
            end;
        end;
    dmdata.tblwinners.close;
    if bfind = true then
        Showmessage('Athlete has been found')
    else
        Showmesage('Athlete has not been found');
    end;
end;

}
procedure TfrmFour.btnEClick(Sender: TObject); (3)
begin
    ledName.show; ✓
    ledSurname.Show;
    ledAge.Show;
    ledEvent.Show; ✓
    LedPosition.Show;
    ledGender.Show;
    ledHouse.Show;
    ledEarnings.Show; ✓
end;

procedure TfrmFour.Button1Click(Sender: TObject); (7)

begin
    with dmData do
        begin
            tblWinners.Last; ✓
            tblWinners.Insert; ✓
            tblWinners['athName'] := ledName.Text;
            tblWinners['athSurname'] := ledSurname.Text; ✓
            tblWinners['Event'] := ledEvent.Text;
            tblWinners['Position'] := ledPosition.Text;
            tblWinners['AgeGroup'] := ledAge.Text; ✓
            tblWinners['Gender'] := ledGender.Text;
            tblWinners['House'] := ledHouse.Text;
            tblWinners['Earning'] := strToint(ledEarnings.Text); ✓
            tblWinners.Post; ✓
        end;
        Showmessage('New record has been added'); ✓
    end;
end.

```

[30]
TOTAAL: 150