



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

SENIORSERTIFIKAAT-EKSAMEN/ NASIONALE SENIORSERTIFIKAAT-EKSAMEN

INLIGTINGSTEGNOLOGIE V1

MEI/JUNIE 2024

NASIENRIGLYNE

PUNTE: 150

Hierdie nasienriglyne bestaan uit 29 bladsye.

ALGEMENE INLIGTING:

- Hierdie nasienriglyne moet as die basis vir die nasiensessie gebruik word. Dit is voorberei om deur nasieners gebruik te word. Daar word na alle nasieners verwag om 'n deeglike standaardiseringsvergadering by te woon om seker te maak dat die riglyne konsekwent geïnterpreteer en tydens die nasien van die leerders se werk toegepas word.
- Let op dat leerders wat 'n alternatiewe korrekte oplossing as wat as voorbeeld van 'n oplossing in die nasienriglyne gegee word verskaf, volle krediet vir die relevante oplossing moet kry tensy die spesifieke instruksies in die vraestel nie gevolg is nie of die vereistes van die vraag nie nagekom is nie.
- **Bylae A, B, C en D** (bladsy 3 tot 11) sluit die nasienriglyne vir elke vraag in.
- **Bylae E, F, G en H** (bladsy 12 tot 29) bevat voorbeelde van oplossings vir Vrae 1 tot 4 in programmeringskode.
- Kopieë van **Bylae A, B, C, D en die opsomming van die leerder se punte** (bladsy 3 tot 11) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word.

BYLAE A

VRAAG 1: NASIENRUBRIEK – ALGEMENE PROGRAMMERINGSVAARDIGHEDE

SENTRUMNOMMER:		EKSAMENNOMMER:		
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT	
1.1	Knoppie [1.1 - Gender] Stel die opskrif ('caption') op "Gender" ✓ Voeg die "Male"-opsie by ✓ Stel die 'Columns'-eienskap op 2 ✓	3		
1.2	Knoppie [1.2 - Random day] Verklaar 'n heelgetal veranderlike vir die ewekansige getal ✓ Genereer ewekansige getal in korrekte reeks (1 – 7) ✓ Vertoon "Day:" ✓ en die ewekansige getal omgeskakel na string ✓ in die opskrif lblQ1_2 ✓ Stel die lysblokkie ('list box') item indeks ✓ op die ewekansige getal ✓ Toets (if) ✓ (ewekansige getal = 1) ✓ of (ewekansige getal = 7) ✓ Vertoon 'Weekend' in edtQ1_2 ✓ Anders ✓ Vertoon 'Weekday' ✓ Alternatief: case lstQ1_2.ItemIndex of (2) 1,7 : edtQ1_2.Text := 'Weekend'; (2) 2..6 : edtQ1_2.Text := 'Weekday'; (2) end;	13		
1.3	Knoppie [1.3 - Calculate] Verklaar 'n geskikte veranderlike/s ✓ Verkry jare uit die spin edit ✓ Bereken bonus: rBonus := Power (P,iJare) ✓ * SQRT ✓ (SQR(P) ✓ / 7 * 20✓) Toets of manager-kontroleblokkie ('checkbox') afgemerkt is ✓ rBonus := rBonus * ✓ 1.1 ✓ OF rBonus := rBonus + rBonus * 0.1 Vertoon bonus in 'n dialoogblokkie ✓ geformateer as 'currency' tot 2 desimale plekke ✓	11		

1.4	<p>Knoppie [1.4 – Title case]</p> <p>Oplossing 1: Gebruik die for.. lus</p> <p>Voeg 'n spasie by aan die einde van die sin ✓ Inisialiseer woord-string ✓ Lus ✓ van 1 tot lengte van sSentence ✓ Toets of die letter in sSentence ✓ is <> aan 'n spasie is nie ✓ Voeg letter van sSentence ✓ by die woord string ✓ anders Skakel die eerste letter ✓ van die word om na hoofletter ✓ Vertoon die woord in redQ1_4 ✓ Maak woord-string ✓ skoon ✓</p> <p>Alternatiewe: Oplossing 2: Gebruik die while.. lus Opossing 3: Gebruik die repeat..until lus Sien voorbeelde in die kode-afdeling</p> <p>Konsepte: Meganisme om die laaste woord in te sluit (1) Inialiseer woord/ teller/ tydelike (1)</p> <p>Lus (1) met korrekte begin, einde limiet/toestand (1)</p> <p>Skei elke woord in twee moontlike metodes (4 punte): <i>Metode 1 (voeg karakters bymekaar):</i> Toets as karakter (1) <> spasie (1) Kopieer karakter (1) tot by woord (1)</p> <p><i>Metode 2 (Kopieer van sin):</i> Toets of karakter (1) = spasie (1) Kopieer woord (1) van indeks 1 tot spasie (1)</p> <p>Skakel die eerste karakter (1) van die woord na 'n hoofletter (1) Vertoon elke woord in redQ4_1 (1)</p> <p>Verwyder woord (1) van indeks 1 tot spasie (1) OF maak woord (1) skoon (1)</p>	13	
	TOTAAL AFDELING A:	40	

BYLAE B

VRAAG 2: NASIENRUBRIEK – DATABASISPROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:		
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT	
2.1	SQL-stellings			
2.1.1	Knoppie [2.1.1 – Free videos]	3		
	SELECT Title, Duration, UploadDate, CreatorID ✓ FROM tblVideos ✓ WHERE FreeVideo = True ✓ AANVAAR ook: WHERE FreeVideo WHERE FreeVideo = Yes WHERE FreeVideo = -1			
2.1.2	Knoppie [2.1.2 – Check domain]	5		
	SELECT CreatorName, Email, Country FROM tblCreators ✓ WHERE NOT Email ✓ LIKE "%@gmail%" ✓ AND ✓ Country = "South Africa" ✓ AANVAAR ook: WHERE email NOT LIKE "%@gmail.com"			
2.1.3	Knoppie [2.1.3 – Latest videos]	4		
	SELECT Top 3 ✓ UploadDate, VideoID, Title FROM tblVideos ✓ ORDER BY UploadDate ✓ DESC ✓			
2.1.4	Knoppie [2.1.4 – Videos per creator]	8		
	SELECT CreatorID, ✓ Count(*) ✓ AS NumberUploaded ✓ FROM tblVideos ✓ GROUP BY ✓ CreatorID ✓ HAVING ✓ Count(*) > 5 ✓ AANVAAR ook: Count(veld naam)			
2.1.5	Knoppie [2.1.5 – Add new creator]	4		
	INSERT INTO ✓ tblCreators ✓ VALUES ✓ ("C011", "TRISHKALOM", "trish@rsmarketing.co.za", "South Africa") ✓			
	Subtotaal:	24		

VRAAG 2: NASIENRUBRIEK – VERVOLG

2.2	Databasismanipulasie		
2.2.1	Knoppie [2.2.1 – Remove creator] Gaan na die eerste rekord in tblCreators ✓ Stap met lus ('loop') deur tblCreators ✓ Toets of tblCreators['CreatorName'] = sCreatorName ✓ Gaan na die eerste rekord in tblVideos ✓ Stap met lus ('loop') deur tblVideos ✓ Toets of (tblCreators ['CreatorID']) = ✓ tblVideos ['CreatorID']) ✓ tblVideos.Delete ✓ anders ✓ tblVideos.Next ✓ Eindig lus (tblVideos) tblCreators.Delete ✓ //eindig toets (if) tblCreators.Next ✓ Eindig lus (tblCreators) NOTE: Lusse hoef nie genes te wees nie. Kan eers die CreatorID kry/vind en die CreatorID gebruik om die rekords van die video's uit die tblVideos te verwyder en dan die skepper uit tblCreators verwyder.	12	
2.2.2	Knoppie [2.2.2 – Change upload date] tblVideos.Edit; ✓ tblVideos ['UploadDate'] ✓ := Date ✓ tblVideos.Post; ✓ AANVAAR ook: DateToStr(Date) DateToStr(DateOf(Now)) FormatDateTime('yyyy/mm/dd', Now())	4	
	Subtotaal:	16	
	TOTAAL AFDELING B:	40	

BYLAE C

VRAAG 3: NASIENRUBRIEK – OBJEK-GEÖRIENTEERDE PROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
3.1.1	Konstruktor-metode: Opskrif met twee parameterwaardes van tipe string ✓ Ken die regte parameterwaardes toe aan fAlbumTitle en fArtist ✓ Ken FALSE toe aan fHighRanking ✓ Ken 0 toe aan fPoints ✓	5	
3.1.2	getPoints-funksie: Funksie opskrif met heelgetalwaarde as terugstuurtipe ✓ fPoints toegeken aan result ✓	2	
3.1.3	updatePoints-prosedure: opskrif van prosedure ✓ met drie heelgetalparameters ✓ fPoints = ✓ albumverkope * 100 ✓ + afgelaaideLiedjies * 10 ✓ + gestroomdeLiedjies ✓	6	
3.1.4	setRanking-prosedure: Opskrif van prosedure met heelgetalparameter ✓ As iNumWeeks > 4 ✓ Stel fHighRanking op true ✓	3	
3.1.5	determineStatus-funksie: Toets of fHighRanking = true ✓ Toets of fPoints >= 5000 AND ✓ fPoints < 10000 ✓ Ken Gold toe aan status ✓ Toets of (fPoints >= 10000) ✓ Ken Platinum toe aan status ✓ Result = status ✓	7	
	Subtotaal: Objekklas	23	

BYLAE D

VRAAG 4: NASIENRUBRIEK – PROBLEEMOPLOSSING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
4.1	<p>Knoppie [4.1 – Sort]</p> <p>Lus ✓ I van 1 tot length(arrPosition) ✓ Reg genes ✓ Lus J van 1 tot length(arrPosition) – 1 ✓ Toets of arrPosition[J] > ✓ arrPosition[J + 1] ✓ iTemp := arrPosition[J]; ✓ arrPosition[J] := arrPosition[J + 1]; ✓ arrPosition[J + 1] := iTemp; ✓</p> <p>ruil arrSongs ✓ by die regte indeks ✓</p> <p>Eindig J lus Eindig I lus</p> <p>Alternatief: for A := 1 to length(arrPosition) - 1 do (2) for B := A+1 to length(arrPosition) do (2) if (arrPosition[A] > arrPosition[B]) then (2) begin iTemp := arrPosition[A]; (1) arrPosition[A] := arrPosition[B]; (1) arrPosition[B] := iTemp; (1) sTemp := arrSongs[A]; arrSongs[A] := arrSongs[B]; arrSongs[B] := sTemp; end; } (2)</p> <p>AANVAAR ook: In plaas van Length(arrPosition) gebruik 20 In plaas van Length(arrPosition) - 1 gebruik 19</p>	11	

4.2	<p>Knoppie [4.2 – New chart]</p> <p>Vertoon opskrifte (Song, Position en Movement) in redQ4 ✓</p> <p>Lêerhantering: AssignFile(tFile, 'Top20.txt') ✓ Reset(tFile) ✓ Lus deur die lêer / Lus I van 1 tot 20 ✓ Lees liedjie uit die teklêer ✓</p> <p>Skuif-veranderinge: Inisialiseer counter J as 0 ✓ Inisialiseer flag as false ✓</p> <p>Lus while J < 20 and Flag = false ✓ Inkrementeer J ✓</p> <p>as liedjie uit tekslêer = arrSongs[J] ✓ stel flag op true ✓</p> <p>as J > I ✓ sMovement = intToStr(J - I) + ' UP' ✓ anders as J < I ✓ sMovement = intToStr(I - J) + ' DOWN' ✓ anders sMovement = 'SAME POSITION' ✓</p> <p>as liedjie nie in die tekslêer gevind is nie / flag = false ✓ sMovement = 'NEW' ✓</p> <p>Vertoon leidjie uit tekslêer, posisie en sMovement in redQ4 ✓</p> <p>Konsepte: Vertoon Vertoon opskrifte in redQ4 (1) Vertoon liedjie, posisie en skuif in redQ4 (1)</p> <p>Lêerhantering AssignFile (1) Reset (1) Lus deur lêer (1) Lees uit lêer (1)</p> <p>Skuif <i>Gebruik tellers/indekse om liedjie se posisie te bepaal: (4 punte):</i> Inisialiseer tellers (1) Lus (1) Inkrement J (1) Toets of liedjie in teks lêer = liedjie in arrSongs (1) Bepaal skuif - up (2) Bepaal skuif - down (2) Bepaal skuif – dieselfde posisie (1) Gebruik 'n vlag (2) om nuwe liedjie te bepaal (2)</p>	19	
	TOTAAL AFDELING D:	30	

OPSOMMING VAN LEERDER SE PUNTE:

SENTRUMNOMMER:		LEERDER SE EKSAMENNOMMER:			
	AFDELING A	AFDELING B	AFDELING C	AFDELING D	
	VRAAG 1	VRAAG 2	VRAAG 3	VRAAG 4	GROOT-TOTAAL
MAKS. PUNTE	40	40	40	30	150
LEERDER SE PUNTE					

BYLAE E: OPLOSSING VIR VRAAG 1

```
//=====
// 1.1 - Gender                                     3 punte
//=====
```

```
procedure TfrmQuestion1.btn1_1Click(Sender: TObject);
begin
  rgpQ1_1.Caption:='Gender';
  rgpQ1_1.Items.Add('Male');
  rgpQ1_1.Columns:=2;
end;
```

```
//=====
// 1.2 - Random day                               13 punte
// =====
```

```
procedure TfrmQuestion1.btnQ1_2Click(Sender: TObject);
var
  iRand: integer;
begin
  iRand := RandomRange(1,8);
  lstQ1_2.ItemIndex:= iRand;
  lblQ1_2.Caption := 'Day: '+IntToStr(iRand);
  if iRand IN [1, 7] then
    begin
      edtQ1_2.Text:= 'Weekend';
    end
  else
    begin
      edtQ1_2.Text:= 'Weekday';
    end;
end;
```

```
//=====
// 1.3 - Calculate                                11 punte
// =====
```

```
procedure TfrmQuestion1.btnQ1_3Click(Sender: TObject);
const
  P = 8;
var
  rBonus:Real;
  iYears:Integer;
begin
  iYears := spnQ1_3.Value;
  rBonus := Power(P,iYears) * Sqrt(Sqr(P) / 7 * 20);
  if chkQ1_3.Checked then
    begin
      rBonus:= rBonus *1.1;
    end;
  ShowMessage(FloatToStrF(rBonus,ffCurrency,10,2));
end;
```

```
//=====
// 1.4 - Title case 13 punte
// =====
```

```
procedure TfrmQuestion1.btnQ1_4Click(Sender: TObject);
var K : integer ;
    sSentence, sWord, sTitleCase : String;
begin
// Provided code
    redQ1_4.Clear;
    sSentence := InputBox('', 'Enter sentence', 'Unlock the power of
technology and ignite innovation');
    // sSentence := InputBox('', 'Enter sentence', 'Let innovation be your
guiding star as you navigate the realms of cyberspace');
```

```
// 1.4 - Title case
// for lus - Oplossing 1
    sSentence := sSentence + ' ';
    sWord := '';
    for K := 1 to Length(sSentence) do
        begin
            if sSentence[K] <> ' ' then
                sWord := sWord + sSentence[K]
            else
                begin
                    sWord[1] := Upcase(sWord[1]);
                    redQ1_4.Lines.Add(sWord);
                    sWord := '';
                end;
            end;
        end;
    end;
```

```
{ while lus - Oplossing 2
while sSentence <> '' do
    begin
        iPosWord := Pos(' ', sSentence);
        if iPosWord > 0 then
            begin
                sWord := Copy(sSentence, 1, iPosWord-1);
                sWord := UPPERCASE(sWord[1]) +
                    Copy(sWord, 2, Length(sWord));
                sSentence := Copy(sSentence, iPosWord+1,
                    Length(sSentence)-iPosWord);
            end
        else
            if iPosWord = 0 then
                begin
                    sWord := UPPERCASE(sSentence[1]) +
                        Copy(sSentence, 2, Length(sSentence));
                    sSentence := '';
                end;
            redQ1_4.Lines.Add(sWord);
        end;
    end; }
```

```
{ repeat..until loop - Oplossing 3

  i := 0;
  repeat
    inc(i);
    if sSentence[i] = ' ' then
  begin
    sTemp := Copy(sSentence, 1, i);
    Delete(sSentence, 1, i);
    sTemp := UpperCase(sTemp[1]) + Copy(sTemp, 2, length(sTemp));
    redQ1_4.Lines.Add(sTemp);
    i := 0;
  end;
until sSentence = ' ';}

end.
```

BYLAE F: OPLOSSING VIR VRAAG 2

```
//
=====
// 2.1 - Afdeling: SQL-stellings
// =====

// =====
// 2.1.1 - Free videos                                     3 punte
// =====

sSQL1 := 'SELECT Title, Duration, UploadDate, CreatorID ' +
        'FROM tblVideos ' +
        'WHERE FreeVideo = True';

// =====
// 2.1.2 - Check domain                                     5 punte
// =====

sSQL2 := 'SELECT CreatorName, Email, Country ' +
        'FROM tblCreators ' +
        'WHERE Email NOT LIKE "%@gmail.com" AND ' +
        'Country = "South Africa"';

// =====
// 2.1.3 - Latest videos                                     4 punte
// =====

sSQL3 := 'SELECT Top 3 UploadDate, VideoID, Title ' +
        'FROM tblVideos ' +
        'ORDER BY UploadDate DESC';

// =====
// 2.1.4 - Videos per creator                               8 punte
// =====

sSQL4 := 'SELECT CreatorID, ' +
        'Count(*) AS NumberUploaded ' +
        'FROM tblVideos ' +
        'GROUP BY CreatorID ' +
        'HAVING Count(*) > 5';

// =====
// 2.1.5 - Add new creator                                   4 punte
// =====

sSQL5 := 'INSERT INTO tblCreators (CreatorID, CreatorName,
        Email, Country) ' +
        'VALUES ("C011", "TRISHKALOM", "trish@rsmarketing.co.za",
        "South Africa");'
```

```
// =====  
// 2.2 - Afdeling: Delphi-kode  
// =====  
  
// =====  
// 2.2.1 - Remove creator 12 punte  
// =====  
  
procedure TfrmQuestion2.btnQ2_2_1Click(Sender: TObject);  
var  
    sCreatorName : String;  
begin  
    // 2.2.1 - Remove creator  
    sCreatorName := cmbQ2_2_1.Text;  
    tblCreators.First;  
    while NOT tblCreators.Eof do  
        begin  
            if tblCreators['CreatorName'] = sCreatorName then  
                begin  
                    tblVideos.First;  
                    while NOT tblVideos.Eof do  
                        begin  
                            if tblCreators['CreatorID'] = tblVideos['CreatorID'] then  
                                tblVideos.Delete  
                            else  
                                tblVideos.Next;  
                            end;  
                        end;  
                    tblCreators.Delete;  
                    end;  
                    tblCreators.Next;  
                end;  
            end;  
        end;  
  
    // Provided code  
    ShowMessage('Records deleted successfully');  
end;  
  
// =====  
// 2.2.2 - Change upload date 4 punte  
// =====  
  
procedure TfrmQuestion2.btnQ2_2_2Click(Sender: TObject);  
begin  
    tblVideos.Edit;  
    tblVideos['UploadDate'] := Date;  
    tblVideos.Post;  
end;
```



```
// =====  
// {$ENDREGION}  
// =====  
// {$REGION 'Provided code: Setup DB connections - DO NOT CHANGE!'}  
// =====  
  
procedure TfrmQuestion2.FormClose(Sender: TObject; var Action:  
TCloseAction);  
begin  
    // Disconnects from database and closes all open connections  
    dbCONN.dbDisconnect;  
end;  
  
procedure TfrmQuestion2.FormShow(Sender: TObject);  
begin  
    // Sets up the connection to database and opens the tables.  
    dbCONN := TConnection.Create;  
    dbCONN.dbConnect;  
    tblManufacturers := dbCONN.tblOne;  
    tblProducts := dbCONN.tblMany;  
    dbCONN.setupGrids(dbgManufacturers, dbgProducts, dbgSQL);  
    pgcDBAdmin.ActivePageIndex := 0;  
end;  
  
// =====  
// {$ENDREGION}  
// =====  
  
end.
```

BYLAE G: OPLOSSING VIR VRAAG 3**Objekklas:**

```
unit Album_U;

interface

uses
    SysUtils, StdCtrls, Dialogs, Math;

type
    TAlbum = class(TObject)
    private
        fAlbumTitle: String;
        fArtist: String;
        fHighRanking: Boolean;
        fPoints: Integer;

    public
// Provided code
        function toString: String;
        function determineStatus: String;
        //
=====

        constructor Create(sAlbumTitle, sArtist: String);
        procedure updatePoints(iAlbumSales, iSongsDownload, iSongsStream:
                                Integer);
        procedure setRanking(iNumWeeks: Integer);
        function getPoints: Integer;
end;

implementation

// =====
// Provided code
// =====
function TAlbum.toString: String;
begin
    Result := 'Title: ' + fAlbumTitle + #13 + 'Artist: ' + fArtist + #13 +
'High ranking: ' +
        BoolToStr(fHighRanking, true) + #13 + 'Number of points: ' +
IntToStr
    (fPoints);
end;
// =====
```

```
// =====
// 3.1.1 Constructor Create                                     5 punte
// =====

constructor TAlbum.Create(sAlbumTitle, sArtist: String);
begin
    fAlbumTitle := sAlbumTitle;
    fArtist := sArtist;
    fHighRanking := false;
    fPoints := 0;
end;

// =====
// 3.1.2 Function getPoints                                    2 punte
// =====

function TAlbum.getPoints: integer;
begin
    Result := fPoints;
end;

// =====
// 3.1.3 Procedure updatePoints                                6 punte
// =====

procedure TAlbum.updatePoints(iAlbumSales, iSongsDownload,
    iSongsStream: Integer);
begin
    fPoints := iAlbumSales * 100 + iSongsDownload * 10 + iSongsStream;
end;

// =====
// 3.1.4 Procedure setRanking                                  4 punte
// =====

procedure TAlbum.setRanking(iNumWeeks: integer);
begin
    if iNumweeks > 4 then
        fHighRanking := true;
end;
```

```
// =====  
// 3.1.5 Function determineStatus                                7 punte  
// =====  
function TAlbum.determineStatus: String;  
var  
    sStatus: String;  
begin  
    // Provided code  
    sStatus := 'None';  
  
    // 3.1.5  
    if (fHighRanking) then  
        if (fPoints >= 5000) AND (fPoints < 10000) then  
            sStatus := 'Gold';  
        if (fPoints >= 10000) then  
            sStatus := 'Platinum';  
    Result := sStatus;  
end;  
  
end.
```

Hoofvormeenheid

```
unit Question3_U;

interface

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

type
  TfrmQuestion3 = class(TForm)
    gbxQ3_2_1: TGroupBox;
    gbxQ3_2_3: TGroupBox;
    redQ3: TRichEdit;
    btnQ3_2_1: TButton;
    gbxQ3_2_2: TGroupBox;
    btnQ3_2_2: TButton;
    Panel1: TPanel;
    Panel2: TPanel;
    btnQ3_2_3: TButton;
    Image1: TImage;
    Label6: TLabel;
    edtQ3_2_1: TEdit;
    Label2: TLabel;
    spnQ3_2_1: TSpinEdit;
    chbQ3_2_1: TCheckBox;
    Label1: TLabel;
    sedQ3_2_2: TSpinEdit;
    procedure btnQ3_2_1Click(Sender: TObject);
    procedure btnQ3_2_2Click(Sender: TObject);
    procedure btnQ3_2_3Click(Sender: TObject);
  private
    const
      arrArtist: array [1 .. 3] of string = ('SZA', 'Morgan Wallen',
                                              'People');
  public

  end;

var
  frmQuestion3: TfrmQuestion3;
  objAlbum: TAlbum;
  iSold, iDownloaded, iStreamed : integer;
implementation

{$R *.dfm}
```

```
// =====
// 3.2.1 Instantiate album object                                5 punte
// =====

procedure TForm1.btnQ3_2_1Click(Sender: TObject);
var
    sAlbum, sArtist: String;
begin
    sAlbum := cmbQ3_2_1.Text;
    sArtist := edtQ3_2_1.Text;
    objAlbum := TAlbum.Create(sAlbum, sArtist);

    // Provided code
    ShowMessage('Album object has been instantiated successfully.');
```

end;

```
// =====
// 3.2.2 Calculate points                                        5 punte
// =====

procedure TfrmQuestion3.btnQ3_2_2Click(Sender: TObject);
begin
    objAlbum.updatePoints(iSold, iDownloaded, iStreamed);
    lblQ3_2_2.Caption := IntToStr(objAlbum.getPoints);
end;
```

```
// =====
// 3.2.3 Set ranking                                           4 punte
// =====

procedure TfrmQuestion3.btnQ3_2_3Click(Sender: TObject);
var
    iNumWeeks : integer;
begin
    iNumWeeks := StrToInt(InputBox('Number of weeks ranked 1', 'Enter
number of weeks', ''));
    objAlbum.setRanking(iNumWeeks);
end;
```

```
// =====
// 3.2.4 Display album details                                  3 punte
// =====

procedure TForm1.btnQ3_2_4Click(Sender: TObject);
begin
    redQ3.Clear;
    redQ3.Lines.Add(objAlbum.toString);
    redQ3.Lines.Add('Status of album: ' +
    objAlbum.determineStatus);end;
```

```
// Provided code - do not change
// =====

procedure TfrmQuestion3.cmbQ3_2_1Change(Sender: TObject);
begin
    edtQ3_2_1.Text := arrArtist[cmbQ3_2_1.ItemIndex + 1];
    iSold := Random(100);
    iDownloaded := Random(500);
    iStreamed := Random(500);

    edtSold.Text := IntToStr(iSold);
    edtDownloaded.Text := IntToStr(iDownloaded);
    edtStreamed.Text := IntToStr(iStreamed);end;
// =====

end.
```

BYLAE H: OPLOSSING VIR VRAAG 4

```
unit Question4_U;

interface
uses
  Windows, Messages, SysUtils, Variants,
  Classes, Graphics,
  Controls, Forms, Dialogs, StdCtrls, ComCtrls,
  ExtCtrls, jpeg, math;
type
  TfrmQuestion4 = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    btnQ4_2: TButton;
    redQ4: TRichEdit;
    btnDisplay: TButton;
    GroupBox1: TGroupBox;
    btnQ4_1: TButton;
    Image1: TImage;
    GroupBox2: TGroupBox;
    GroupBox3: TGroupBox;
    procedure btnQ4_2Click(Sender: TObject);
    procedure btnDisplayClick(Sender: TObject);
    procedure btnQ4_1Click(Sender: TObject);
    procedure FormShow(Sender: TObject);
  private
    procedure DisplayArrays;
    { Private declarations }
  public
    { Public declarations }
  end;
var
  frmQuestion4: TfrmQuestion4;

  arrSongs: array [1 .. 20] of String = (
    'Castle of Hope', 'Deep Green Hills', 'Backseat Kiss', 'Earning
    Nocturno', 'Edges of Dawing', 'Free Future', 'Heart Hymn', 'Heroic
    Flavor', 'Me and You', 'New York Dirt', 'Not Night', 'Adagio', 'Running
    Study', 'So Hard Spring', 'Sound of Illusion', 'The Celebration',
    'Unexpected Skies', 'Wait for Friends', 'Warm Heart', 'Winter Friends');

  arrPosition: array [1 .. 20] of integer = (
    4, 6, 11, 20, 2, 12, 19, 5, 1, 10, 14, 13, 17, 9, 3, 18, 16, 7, 8,
    15);

implementation

{$R *.dfm}
```



```
// =====  
// 4.1 - Sort 11 punte  
// =====  
  
procedure TfrmQuestion4.btnQ4_1Click(Sender: TObject);  
var  
    I, iTemp: integer;  
    J: integer;  
    sTemp: String;  
begin  
    // Provided code  
    redQ4.Clear;  
  
    // Question 4.1  
  
    for I := 1 to length(arrPosition) do  
    begin  
        for J := 1 to length(arrPosition) - 1 do  
        begin  
            if arrPosition[J] > arrPosition[J + 1] then  
            begin  
                iTemp := arrPosition[J];  
                arrPosition[J] := arrPosition[J + 1];  
                arrPosition[J + 1] := iTemp;  
  
                sTemp := arrSongs[J];  
                arrSongs[J] := arrSongs[J + 1];  
                arrSongs[J + 1] := sTemp;  
  
            end;  
        end;  
    end;  
  
    DisplayArrays;  
  
end;
```

```
// =====  
// 4.2 New chart                                     19 punte  
// =====  
  
procedure TfrmQuestion4.btnQ4_2Click(Sender: TObject);  
var  
    tFile: TextFile;  
    J, iNew: integer;  
    bFound: boolean;  
    sNewSong, sMsg: String;  
  
// Variables for Alternative 2 and 3  
// tFile: TextFile;  
// arrNewSongs: array [1 .. 20] of String;  
// I: integer;  
// J, iDiff, iPos: integer;  
// bFlag, bFound: boolean;  
// sLine, sOutput: String;  
begin  
    // Question 4.2  
  
    redQ4.Clear;  
    AssignFile(tFile, 'Top20.txt');  
    Reset(tFile);  
  
    redQ4.lines.add('Song' + #9 + 'Position' + #9 + 'Movement');  
  
    for iNew := 1 to 20 do  
        begin  
            readln(tFile, sNewSong);  
            J := 0;  
            bFound := false;  
            while (J < 20) AND (NOT bFound) do  
                begin  
                    inc(J);  
                    if sNewSong = arrSongs[J] then  
                        begin  
                            bFound := true;  
                            if J > iNew then  
                                sMsg := IntToStr(J - iNew) + ' UP'  
                            else if iNew > J then  
                                sMsg := IntToStr(iNew - J) + ' DOWN'  
                            else  
                                sMsg := 'SAME POSITION';  
                            end;  
                        end;  
                    if NOT bFound then  
                        sMsg := 'NEW';  
  
                    redQ4.lines.add(sNewSong + #9 + IntToStr(iNew) + #9 + sMsg);  
  
                end;  
            end;  
        end;  
    end;
```

```
{ //Alternative 2
redQ4.Clear;

AssignFile(tFile, 'Top20.txt');

try
    reset(tFile);

    for I := 1 to 20 do
        begin

            readln(tFile, arrNewSongs[I]);

        end;
redQ4.lines.add('Song' + #9 + 'Position' + #9 + 'Movement');

for I := 1 to length(arrNewSongs) do
    begin

        J := 0;
        bFlag := true;
        while (J < 20) AND (bFlag) do
            begin
                inc(J);
                if arrNewSongs[I] = arrSongs[J] then
                    begin
                        if J - I > 0 then
                            begin
                                sLine := arrNewSongs[I] + #9 + intToStr(I) + #9 + '(' +
                                    intToStr(abs(J - I)) + ' UP)';
                            end
                        else if J - I < 0 then
                            Begin
                                sLine := arrNewSongs[I] + #9 + intToStr(I) + #9
                                    + '(' + intToStr(abs(J - I)) + ' DOWN)';
                            End
                        else
                            begin
                                sLine := arrNewSongs[I] + #9 + intToStr(I) + #9
                                    + '(SAME POSITION)';
                            end;
                        bFlag := false;
                    end
                else
                    begin
                        sLine := arrNewSongs[I] + #9 + intToStr(I) + #9 + '(NEW)';
                    end;
                end;
                redQ4.lines.add(sLine);
            end;
        except
            ShowMessage('File not found');
            Application.Terminate;
        end;
    }
}
```

```
{ //Alternative 3
AssignFile(tFile, 'Top20.txt');
Reset(tFile);
j:=0;
while not eof(tFile) do
Begin
  Readln(tFile, sLine);
  bFlag:= False;
  Inc(j);
  i:=0;
  sOutput:= sLine+#9+IntToStr(j);
  while(bFlag = false) AND (i<20) do
  begin
    Inc(i);
    if arrSongs[i] = sLine then
    begin
      bFlag := True;

      iDiff := i - j;
      if iDiff < 0 then
      begin
        sOutput := sOutput+ #9+'('+IntToStr(ABS(iDiff))+ ' DOWN) ' ;
      end
      else if iDiff > 0 then
      begin
        sOutput := sOutput+ #9+'('+IntToStr(iDiff)+ ' UP) ' ;
      end
      else if iDiff = 0 then
      begin
        sOutput := sOutput+ #9+'SAME POSITION';
      end ;

    end; //if bflag - true
  end; //while for array

  if bFlag = False then
  begin
    sOutput:= sOutput+#9+'NEW';
  end;

  redQ4.Lines.Add(sOutput);
End; //while for file
CloseFile(tFile);
end; }
```

```
// =====  
// Provided code  
// =====  
procedure TfrmQuestion4.FormShow(Sender: TObject);  
begin  
    redQ4.Paragraph.TabCount := 3;  
    redQ4.Paragraph.Tab[0] := 0;  
    redQ4.Paragraph.Tab[1] := 120;  
    redQ4.Paragraph.Tab[2] := 180;  
end;  
  
procedure TfrmQuestion4.DisplayArrays;  
var  
    I: Integer;  
begin  
    // Provided code  
    redQ4.lines.add('TOP CHARTS');  
    redQ4.lines.add(format('%-20s%-15s%-5s', ['Songs', 'Artist',  
'Position']));  
    for I := 1 to length(arrSongs) do  
        redQ4.lines.add(format('%-20s%-15s%-5d', [arrSongs[I],  
arrArtists[I], arrPosition[I]]));  
    end;  
  
end.  
//=====
```

End of provided code

```
//=====
```