# INLIGTINGSTEGNOLOGIE V1

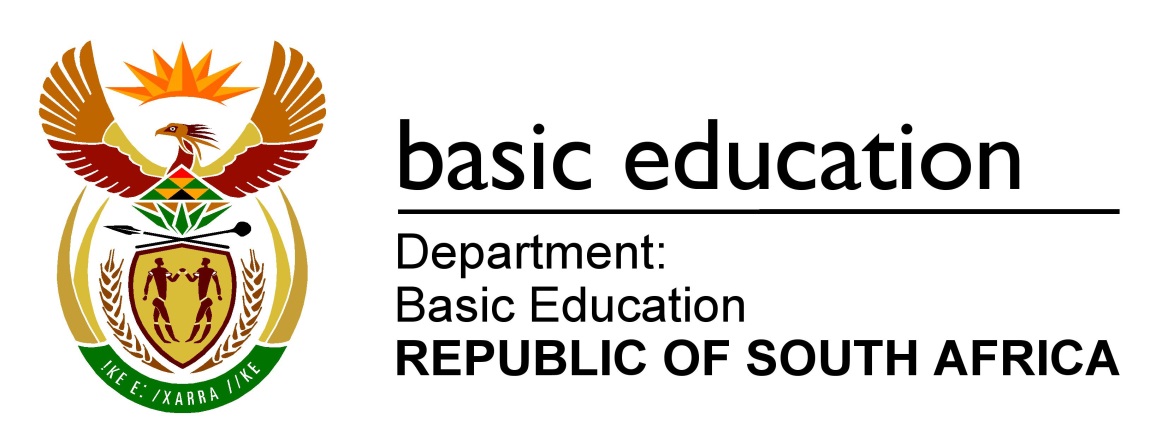
**FEBRUARIE/MAART 2018**

**NASIENRIGLYNE**

# NASIONALE

# SENIOR SERTIFIKAAT

# GRAAD 12



**PUNTE: 150**

**Hierdie nasienriglyne bestaan uit 21 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 kandidate 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. |  |  |
|  |  |  |
| * **Bylaag A, B en C** (bladsy 3–8) sluit die nasienrubriek vir elke om te gebruik vir enigeen van die twee programmeringstale in. |  |  |
|  |  |  |
| * **Bylaag D, E en F** (bladsy 9–21) bevat voorbeelde in programmeringskode van oplossings vir VRAAG 1 tot VRAAG 3. |  |  |
|  |  |  |
| * Kopieë van **Bylaag A, B en C** (bladsy 3–8) en **die opsomming van leerderpunte** (bladsy 8) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word. |  |  |

**BYLAAG A**

**AFDELING A**

**VRAAG 1: NASIENRUBRIEK - ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SENTRUMNOMMER: | | EKSAMENNOMMER: | | |
| **VRAAG** | **BESKRYWING** | | MAKS. PUNTE | LEERDER- PUNT |
|  | ***'n Leerder moet slegs een maal gepenaliseer word as dieselfde fout herhaal word.*** | |  |  |
| 1.1 | **[Knoppie 1.1 – Totale area]**  Verkry die radius ✓ skakel om na real✓  Verkry die basis en hoogte ✓ uit editblokkies  Area van sirkel = pi \* ✓ Sqr(radius) ✓  Area van driehoeke = 0.5 \* basis \* hoogte ✓ \* aantal driehoeke ✓  Vertoon die area van sirkel✓ met byskrif ✓  Vertoon die area van driehoeke in totaal ✓  Bereken totale area van sirkel en driehoeke ✓  Vertoon die totale area tot 2 desimale plekke ✓ | | **12** |  |
| 1.2 | **[Knoppie 1.2 – Volgende blou maan]**  Vind posisie van dubbelpunt (':') in byskrif ('label') ✓  Verkry maanjaar vanuit byskrif ✓ en skakel om na integer ✓  Repeat  Tel 3 by✓ die maanJaar ✓  Until maanJaar > ✓ jaar van huidige stelsel datum  Vertoon maanJaar ✓  Korrekte lus bv. repeat...until ✓ | | **9** |  |
| 1.3 | **[Knoppie 1.3 – Grootste gemene deler]**  Verkry getal 1 van editblokkie ✓ skakel om na integer✓  Verkry getal 2 van editblokkie ✓  Lus x van 1 ✓ tot Minimum (getal1,getal2) ✓  Toets of (getal 1 mod x = 0) ✓ AND ✓ (getal2 mod x =0) ✓  Stel ggd na x ✓  Vertoon ggd in die editblokkie ✓ | | **10** |  |
| 1.4 | **Knoppie [1.4 – Verwyder klinkers]**  Verkry sin vanuit editblokkie ✓  Stel 'n tydelike veranderlike na die eerste karakter van sin✓  Lus x van 2 ✓ tot lengte van sin ✓  if (sin[x - 1] = ' ') ✓ OR ✓  NOT(upcase✓ (sin[x]) ✓ in ['A', 'E', 'I', 'O', 'U']) ✓  tydelik = tydelik + sin[x] ✓  Vertoon tydelik in editblokkie ✓ | | **11** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 1.5 | **[Knoppie 1.5 – Skyfievertoning]**  Vir inisialisering van totaal en groeptotaal ✓  Stel totaal = 0  Stel groepNommer= 0  Gebruik 'n lus ✓en toets of  totaal < SitplekkeBeskikbaar ✓ (konstante gegee)  Gebruik 'n toevoerblokkie ('input box') om die getal mense in te lees ✓  Indien getal mense in groep ✓  <= (SitplekkeBeskikbaar - total)✓  Inkrementeer groepNommer ✓  Inkrementeer totaal met die getal mense ✓  Vertoon groepNommer and getal mense ✓  Anders✓  Vertoon boodskap ✓ met korrekte waardes ✓  Einde van lus | **12** |  |
|  | **TOTAAL AFDELING A:** | **54** |  |

**BYLAAG B**

**AFDELING B**

**VRAAG 2: NASIENRUBRIEK - OBJEK-GEÖRIENTEERDE PROGRAMMERING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SENTRUMNOMMER: | | EKSAMENNOMMER: | | |
| **VRAAG** | **BESKRYWING** | | **VRAAG** | **LEERDER-PUNT** |
| 2.1.1 | **Konstruktor:**  Opskrif met SLEGS vier waardes✓  Drie string parameters en een real ✓  Ken parameter waardes toe aan vier attribute✓  Ken false toe aan fNavigasieStatus attribuut✓ | | **4** |  |
| 2.1.2 | **toegangsMETODE:**  Konstellasie: 'return' tipe string✓ stuur attribuut terug✓ | | **2** |  |
| 2.1.3 | **stelNavigasieStatus** **PROSEDURE:**  Gebruik van 'n prosedure✓ (nie funksie nie)  Ontvang Boolse waarde✓  Stel waarde van fNavigasieStatus attribuut✓ | | **3** |  |
| 2.1.4 | **bepaalSigbaarheid FUNKSIE:**  Indien afstand kleiner as 80✓  Duidelik sigbaar✓  Anders✓ {afstand *>= 80*}  Indien afstand <= 900 ✓  Indien grootte <= 2 ✓  Skaars sigbaar met blote oog ✓  Anders {*grootte* *>2*}✓  Sigbaar met behulp van standaard optiese hulp ✓  Anders {*afstand > 900*}✓  Slegs sigbaar met behulp van gespesialiseerde  optiese hulp ✓  Stel resultaat na sigbaarheid waarde✓ | | **11** |  |
| 2.1.5 | **toString** **METODE**  Voeg attribute toe tot afvoer string - naam, konstellasie, grootte en ligjare ✓ met korrekte byskrifte ✓  Indien ster 'n navigasie ster is✓  Voeg by sternaam ' is 'n navigasie ster.' by boodskap ✓  Anders  Voeg by sternaam ' is 'n passiewe ster.' by boodskap ✓  Stel resultaat na saamgestelde sin ✓ | | **6** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 2.2.1 | **Knoppie – [2.2.1 – Instansieer objek]:**  Verkry die naam van die ster vanuit die kombinasieblokkie ✓  Assign ✓ en Reset ✓  Inisialiseer Vlag ('found')/Teller ✓  Voorwaardelike lus (while/repeat) NOT EOF✓  & NOT Found✓  Lees reël vanuit die tekslêer ✓  Vergelyk of reël = naam van die ster ✓  Lees DRIE reëls vanuit die tekslêer ✓  Instansieer ster objek - Create objSterX ✓ (param) ✓  Korrekte getal en volgorde van parameters✓  objSterX := TStar.Create(parameters)  Boolse veranderlike gestel na false om deur skikking  te soek ✓  Lus van 1 tot lengte van skikking✓  Toets of sternaam voorkom in skikking ✓✓  Stel Boolse verandelike na true✓  Roep 'set' metode vir navigasie ster, ✓ deur resultaat  van Boolse veranderlike as parameter te gebruik✓  Verander vlag na true ✓  Aktiveer ('enable') die paneel pnlKnoppies ✓  Toets of ster nie gevind is nie ('NOT found'):  Vertoon boodskap ✓  Aktiveer tab sheet 2 ✓  Sluit lêer (CloseFile)✓ | **24** |  |
| 2.2.2 | **Knoppie – [2.2.2 - Vertoon]:**  Gebruik toString metode en vertoon objek✓  Laai konstellasie prent vanaf lêer ✓en vertoon✓ | **3** |  |
| 2.2.3 | **Knoppie – [2.2.3 - Sigbaarheid]:**  Gebruik die sterobjek om metodes te roep:  objSter.getNaam✓  objSter.bepaalSigbaarheid✓  Vertoon in korrekte formaat ✓ | **3** |  |
|  | **TOTAAL AFDELING B:** | **56** |  |

**BYLAAG C**

**AFDELING C**

**VRAAG 3: NASIENRUBRIEK - PROBLEEMOPLOSSING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SENTRUMNOMMER: | | EKSAMENNOMMER: | | |
| **VRAAG** | **BESKRYWING** | | **VRAAG** | **LEERDER-PUNT** |
| 3.1 | **Knoppie [3.1 – Begin spel]**  Lees die vlak uit die radiogroep✓  Toets vir vlak ✓en ken getal planete toe aan veranderlike ✓  Vlak 1 (50); Vlak 2 (40); Vlak 3 (30)  Vertoon 0 op paneel en stel ItemIndeks van cmbRy en cmbKol op 0 ✓  Vee afvoerarea met foutiewe raaiskote uit✓  Inisialiseer veranderlikes vir tellers ✓  *VulOp ('Populate')*:  Vul skikking met - ✓deur lus te gebruik vir rye en kolomme ✓  Herhaal korrekte aantal kere (while/repeat) ✓ om:  Lukrake ('random') posisie vir ry ✓ en kolom te bepaal ✓  Toets of posisie planeet bevat ✓ en..  plaas # in posisie✓  verminder planete wat geplaas moet word ✓  *Vertoon*:  Skoon spelbord in afvoerarea ✓  Lus deur die rye ✓  Skep afvoerstring ✓  Lus deur die kolomme✓  Indien planeet voeg #✓ by tot afvoerstring  Anders voeg -✓ by afvoerstring  Vertoon afvoerstring op spelbord✓  Aktiveer Speel Knoppie ✓ | | **22** |  |
| 3.2 | **Knoppie [3.2 - Speel]**  Verkry ry en kolom uit kombinasieblokkie ✓  Inkrementeer getal raaiskote ✓  Vind karakter by die posisie in skikking✓  Toets of karakter planeet is ✓  Vervang met plekhouer✓  Anders✓  Vertoon ry ✓en kolom ✓in area vir foutiewe raaiskote  Dateer vertoon op spelbord op ✓  Vertoon getal raaiskote op paneel ✓  Indien twee planete gevind is ✓  vertoon 'Spel gewen' boodskap ✓  Anders  vertoon 'Spel verloor' boodskap. ✓ | | **13** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 3.3 | **Knoppie [3.3 – Onthul planete]**  Lus deur rye✓  Skep afvoerstring✓  Lus deur kolomme✓  Voeg skikkingwaarde by afvoerstring✓  Vertoon afvoerstring✓ | **5** |  |
|  | **TOTAAL AFDELING C:** | **40** |  |

**OPSOMMING VAN LEERDER PUNTE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SENTRUMNOMMER: | | EKSAMENNOMMER: | | |
|  | **AFDELING A** | **AFDELING B** | **AFDELING C** |  |
|  | **VRAAG 1** | **VRAAG 2** | **VRAAG 3** | **GROOT-TOTAAL** |
| **MAKS.**  **PUNTE** | **54** | **56** | **40** | **150** |
| **LEERDER**  **PUNTE** |  |  |  |  |

**BYLAAG D: OPLOSSING VIR VRAAG 1**

unit Vraag1\_U;

interface

uses

Windows,Messages,SysUtils,Variants,Classes, Graphics, Controls,Forms,

Dialogs, StdCtrls, ExtCtrls, ComCtrls, jpeg, DateUtils, Math;

type

TfrmV1 = class(TForm)

PageControl1: TPageControl;

TabSheet1: TTabSheet;

TabSheet2: TTabSheet;

Prent1: TImage;

btnV1\_1: TButton;

redV1\_1: TRichEdit;

edtRadius: TEdit;

edtBasis: TEdit;

edtHoogte: TEdit;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label7: TLabel;

lblInfo: TLabel;

Label9: TLabel;

edtVr1\_2: TEdit;

btnV1\_2: TButton;

Prent2: TImage;

TabSheet4: TTabSheet;

TabSheet5: TTabSheet;

TabSheet6: TTabSheet;

edtSin: TEdit;

Label10: TLabel;

btnV1\_4: TButton;

edtVerwyderKlinkers: TEdit;

Label11: TLabel;

Label12: TLabel;

edtGetal1: TEdit;

edtGetal2: TEdit;

btnV1\_3: TButton;

edtGGD: TEdit;

redVraag1\_5: TRichEdit;

btnV1\_5: TButton;

procedure btnV1\_1Click(Sender: TObject);

procedure btnV1\_2Click(Sender: TObject);

procedure btnV1\_3Click(Sender: TObject);

procedure btnV1\_5Click(Sender: TObject);

procedure btnV1\_4Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

frmV1: TfrmV1;

implementation

{$R \*.dfm}

// Vraag 1.1

// ========================================================================

procedure TfrmV1.btnV1\_1Click(Sender: TObject);

Const

rPi = 3.14159;

iGetal = 8;

Var

rRadius, rAreaSirkel, rBasis, rHoogte, rAreaDriehoeke, rTotaleArea: real;

begin

rRadius := StrToFloat(edtRadius.Text);

rBasis := StrToFloat(edtBasis.Text);

rHoogte := StrToFloat(edtHoogte.Text);

rAreaSirkel := rPi \* Sqr(rRadius);

redV1\_1.Lines.Add('Area van sirkel = ' + FloatToStr(rAreaSirkel));

rAreaDriehoeke := 0.5 \* rBasis \* rHoogte\* iGetal;

redV1\_1.Lines.Add('Totale area van driehoeke = ' +

FloatToStr(rAreaDriehoeke));

rTotaleArea := rAreaSirkel+ rAreaDriehoeke;

redV1\_1.Lines.Add('Totale area = '

+ FloatToStrF(rTotaleArea, ffFixed, 6, 2));

end;

// Vraag 1.2

// ========================================================================

procedure TfrmV1.btnV1\_2Click(Sender: TObject);

Const

iToename = 3;

Var

dDatum: TDateTime;

iDubPunt, iMaanJaar: integer;

begin

iDubPunt := pos(':', lblInfo.Caption);

iMaanJaar := StrToInt(copy(lblInfo.Caption, iDubPunt + 1, 4));

repeat

iMaanJaar := iMaanJaar + iToename;

until iMaanJaar > YearOf(now);

edtVR1\_2.Text := IntToStr(iMaanJaar);

end;

// Vraag 1.3

// ========================================================================

procedure TfrmV1.btnV1\_3Click(Sender: TObject);

Var

iGetal1, iGetal2, iGGD, i: integer;

begin

iGetal1 := StrToInt(edtGetal1.Text);

iGetal2 := StrToInt(edtGetal2.Text);

for i := 1 to Min(iGetal1, iGetal2) do

begin

if (iGetal1 mod i = 0) and (iGetal2 mod i = 0) then

begin

iGGD := i;

end;

end;

edtGGD.Text := IntToStr(iGGD);

end;

// Vraag 1.4

// ========================================================================

procedure TfrmV1.btnV1\_4Click(Sender: TObject);

Var

sSin, sTyd: String;

i: integer;

begin

sSin := edtSin.Text;

sTyd := sSin[1];

for i := 2 to length(sSin) do

begin

if (sSin[i - 1] = ' ') OR

NOT(upcase(sSin[i]) in ['A','E','I','O','U']) then

sTyd := sTyd + sSin[i];

end;

edtVerwyderKlinkers.Text := sTyd;

end;

// Vraag 1.5

// ========================================================================

procedure TfrmV1.btnV1\_5Click(Sender: TObject);

Const

iSitplekkeBeskikbaar = 100;

Var

iGroepNommer, iGetalMense, iTotaal: integer;

begin

// Kode wat voorsien is

redV1\_5.Clear;

redV1\_5.Lines.Add('Groepnommer' + #9#9 + 'Getal mense');

iTotaal := 0;

iGroepNommer := 0;

while iTotaal < iSitplekkeBeskikbaar do

begin

iGetalMense := StrToInt(Inputbox('',

'Voer die getal mense in die groep in', ''));

if iGetalMense <= (iSitplekkeBeskikbaar - iTotaal) then

begin

Inc(iGroepNommer);

Inc(iTotaal, iGetalMense);

redQues1\_5.Lines.Add(IntToStr(iGroepNommer) + #9#9#9 + IntToStr

(iGetalMense));

end // if

else

begin

ShowMessage('Kan nie ''n groep van ' + IntToStr(iGetalMense)

+ ' mense aanvaar nie.' + #13

+ 'Getal sitplekke beskikbaar is ' + IntToStr(100 - iTotaal));

end; // else

end; // while

end;

end.

**BYLAAG E: OPLOSSING VIR VRAAG 2**

**OBJEKKLAS:**

unit Ster\_U;

interface

type

TSter = class(TObject)

private

// Kode wat voorsien is

fNaam: String;

fGrootte: real;

fAfstand: integer;

fKonstellasie: String;

fNavigasieStatus: Boolean;

public

constructor Create(sNaam: String; rGrootte: real; iAfstand: integer;

sKonstellasie: String);

function getKonstellasie: String;

procedure setNavigasieStatus(bStatus: Boolean);

function bepaalSigbaarheid: String;

function toString: String;

// Kode wat voorsien is

function getNaam: String;

end;

implementation

Uses Math, SysUtils;

{$R+}

========================================================================

// Vraag 2.1.1

========================================================================

constructor TSter.Create(sNaam: String; rGrootte: real; iAfstand: integer; sKonstellasie: String);

begin

fNaam := sNaam;

fGrootte := rGrootte;

fAfstand := iAfstand;

fKonstellasie := sKonstellasie;

fNavigasieStatus := false;

end;

========================================================================

// Vraag 2.1.2

========================================================================

function TSter.getKonstellasie: String;

begin

result := fKonstellasie;

end;

========================================================================

// Vraag 2.1.3

========================================================================

procedure TSter.setNavigasieStatus(bStatus: Boolean);

begin

fNavigasieStatus := bStatus;

end;

========================================================================

// Vraag 2.1.4

========================================================================

function TSter.bepaalSigbaarheid: String;

var

sSigbaarheid: String;

begin

if (fAfstand < 80) then

sSigbaarheid := 'Duidelik sigbaar'

else if (fAfstand <= 900) then

if (fGrootte <= 2) then

sSigbaarheid := 'Skaars sigbaar met die blote oog'

else sSigbaarheid := 'Slegs sigbaar met behulp van standaard optiese hulp'

else sSigbaarheid := 'Slegs sigbaar met behulp van gespesialiseerde optiese hulp';

Result := sSigbaarheid;

end;

========================================================================

// Vraag 2.1.5

========================================================================

function TSter.toString: String;

var

sAfvoer: String;

begin

sAfvoer := Format('%s behoort aan die' + ' %s-konstellasie.' + #13 + #13 +

'Die ster het ''n grootte van %3.2f en is %d ligjare vanaf die Aarde.' +

#13+ #13 , [fNaam, fKonstellasie, fGrootte, fAfstand]);

if fNavigasieStatus then

sAfvoer := sAfvoer + fNaam + ' is ''n navigasie ster.'

else

sAfvoer := sAfvoer + fNaam + ' is ''n passiewe ster.';

result := sAfvoer;

end;

========================================================================

// Kode wat voorsien is

// ========================================================================

function TSter.getNaam: String;

begin

result := fNaam;

end;

end.

**HOOFVORM EENHEID: VRAAG2\_U.PAS**

unit Vraag2\_U;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls, Buttons, ComCtrls, ExtCtrls, jpeg, Ster\_U;

type

TfrmV2 = class(TForm)

bmbClose: TBitBtn;

btnV2\_2\_1: TButton;

redAfvoer: TRichEdit;

lblOpskrif: TLabel;

cmbSter: TComboBox;

imgV2: TImage;

pnlKnoppies: TPanel;

lblVrNom: TLabel;

btnV2\_2\_2: TButton;

btnV2\_2\_3: TButton;

procedure FormCreate(Sender: TObject);

procedure FormCanResize(Sender: TObject; var NewWidth, NewHeight: Integer;

var Resize: Boolean);

procedure cmbSterChange(Sender: TObject);

procedure btnV2\_2\_1Click(Sender: TObject);

procedure btnV2\_2\_2Click(Sender: TObject);

procedure btnV2\_2\_3Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

frmV2: TfrmV2;

// Kode wat voorsien is

objSterX: TSter;

implementation

var

arrNavigasieSterre: array [1 .. 58] of String = (

'Alpheratz','Ankaa','Schedar','Diphda','Achernar','Hamal','Acamar', 'Menkar','Mirfak','Aldebaran','Rigel','Capella','Bellatrix','Elnath', 'Alnilam','Betelgeuse','Canopus','Sirius','Adhara','Procyon','Pollux', 'Avior','Suhail','Miaplacidus','Alphard','Regulus','Dubhe','Denebola', 'Gienah','Acrux','Gacrux','Alioth','Spica','Alkaid','Hadar','Menkent',

'Rigil Kentaurus','Arcturus','Zubenelgenubi','Kochab','Alphecca',

'Antares','Atria','Sabik','Shaula','Rasalhague','Eltanin',

'Kaus Australis','Vega','Nunki','Altair','Peacock','Deneb','Enif',

'Al Na''ir','Fomalhaut','Markab','Polaris');

{$R \*.dfm}

{$R+}

// Vraag 2.2.1

// =====================================================================

procedure TfrmV2.btnV2\_2\_1Click(Sender: TObject);

var

sSterNaam: String;

tLêer: textFile;

sLyn, sKonstelasie: String;

rGrootte : Real;

bLêerGevind, bSkikkingGevind: Boolean;

A, iAfstand : Integer;

begin

// Vraag 2.2.1

sSterNaam := cmbSter.Items[cmbSter.ItemIndex];

AssignFile(tLêer, 'SterData.txt');

Reset(tLêer);

bLêerGevind := false;

While NOT Eof(tLêer) AND NOT bLêerGevind do

begin

Readln(tLêer, sLyn);

if Trim(sSterNaam) = Trim(sLyn) then

begin

Readln(tLêer, rGrootte);

Readln(tLêer, iAfstand);

Readln(tLêer, sKonstelasie);

objSterX := TSter.Create(sSterNaam, rGrootte, iAfstand,

sKonstelasie);

bSkikkingGevind := false;

for A := 1 to Length(arrNavigasieSterre) do

if pos(arrNavigasieSterre[A], objSterX.getNaam) > 0 then

bSkikkingGevind := true;

objSterX.setNavigasieStatus(bSkikkingGevind);

bLêerGevind := true;

pnlKnoppies.Show;

end;

end;

if NOT bLêerGevind then

begin

sLyn := 'Die ster is nie in die lêer gevind nie.';

MessageDlg(sLyn, mtError, [mbOK], 0);

// pctrlQ2.ActivePageIndex := 1;

pnlKnoppies.Hide;

end;

CloseFile(tLêer);

end;

// Vraag 2.2.2

// =====================================================================

procedure TfrmV2.btnV2\_2\_2Click(Sender: TObject);

begin

// Vraag 2.2.2

redAfvoer.Clear; //Kode wat voorsien is

redAfvoer.Lines.Add(objSterX.toString);

imgV2.Picture.LoadFromFile(objSterX.getKonstellasie + '.jpg');

end;

// =====================================================================

// Vraag 2.2.3

// =====================================================================

procedure TfrmV2.btnV2\_2\_3Click(Sender: TObject);

begin

// Vraag 2.2.3

redAfvoer.Clear;

redAfvoer.Paragraph.TabCount := 1;

redAfvoer.Paragraph.Tab[0] := 50;

redAfvoer.Lines.Add('Ster: ' + #9#9 +objSterX.getNaam);

redAfvoer.Lines.Add('Sigbaarheid: ' +#9 + objSterX.bepaalSigbaarheid);

end;

{$REGION ' Kode wat voorsien is'}

// =====================================================================

// Kode wat voorsien is - Moenie verander nie

// =====================================================================

procedure TfrmV2.cmbSterChange(Sender: TObject);

begin

redAfvoer.Clear;

pnlKnoppies.Hide;

imgV2.Picture := nil;

end;

procedure TfrmV2.FormCanResize(Sender: TObject;

var NewWidth, NewHeight: Integer; var Resize: Boolean);

begin

Resize := false;

end;

procedure TfrmV2.FormCreate(Sender: TObject);

begin

CurrencyString := 'R'; // pctrlQ2.ActivePageIndex := 0;

pnlKnoppies.Hide;

end;

{$ENDREGION}

end.

**BYLAAG F: OPLOSSING VIR VRAAG 3**

unit Vraag3\_U;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls, ExtCtrls, ComCtrls, Buttons;

type

TfrmV3 = class(TForm)

redV3Spelbord: TRichEdit;

rgbV3: TRadioGroup;

btnV3\_1BeginSpel: TButton;

btnClose: TBitBtn;

btnV3\_2Speel: TButton;

cmbRy: TComboBox;

cmbKol: TComboBox;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

redV3Foutief: TRichEdit;

btnV3\_3Onthull: TButton;

Label5: TLabel;

pnlV3GetalRaaiskote: TPanel;

pnlSpeel: TPanel;

procedure btnV3\_1BeginSpelClick(Sender: TObject);

procedure VulOp;

procedure vertoon;

procedure btnV3\_2SpeelClick(Sender: TObject);

procedure btnV3\_3OnthullClick(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

frmV3: TfrmV3;

iAantal: integer;

iGevind: integer = 0;

iGetal: integer;

// ===================================================================

// Kode wat voorsien is

// ===================================================================

arrSpel: array [1 .. 9, 1 .. 9] of char;

implementation

{$R \*.dfm}

{$R+}

// ===================================================================

// Vraag 3.1

// ===================================================================

procedure TfrmV3.btnV3\_1BeginSpelClick(Sender: TObject);

var

iVlak: integer;

begin

// Vraag 3.1

iVlak := rgbV3.ItemIndex;

case iVlak of

0:

iAantal := 50;

1:

iAantal := 40;

2:

iAantal := 30;

end;

iGevind := 0;

iGetal := 0;

pnlV3GetalRaaiskote.Caption := IntToStr(iGevind);

btnV3\_2Speel.Enabled := true;

redV3Foutief.Clear;

cmbRy.ItemIndex:=0;

cmbKol.ItemIndex:=0;

VulOp;

vertoon;

end;

procedure TfrmV3.VulOp;

var

iRy, iKol: integer;

begin

for iRy := 1 to Length(arrSpel) do

begin

for iKol := 1 to Length(arrSpel) do

begin

arrSpel[iRy, iKol] := '-';

end;

end;

while iAantal <> 0 do

begin

iRy := random(9) + 1;

iKol := random(9) + 1;

if (arrSpel[iRy, iKol] = '-') then

begin

arrSpel[iRy, iKol] := '#';

dec(iAantal);

end;

end;

end;

procedure TfrmV3.vertoon;

var

iRy, iKol: integer;

sReel: string;

begin

redV3Spelbord.Clear;

for iRy := 1 to Length(arrSpel) do

begin

sReel := '';

for iKol := 1 to Length(arrSpel) do

begin

if arrSpel[iRy, iKol] = '$' then

sReel := sReel + '# '

else

sReel := sReel + '- ';

end;

redV3Spelbord.Lines.Add(sReel);

end;

end;

// ===================================================================

// Vraag 3.2

// ===================================================================

procedure TfrmV3.btnV3\_2SpeelClick(Sender: TObject);

var

iRy, iKol: integer;

cKar: char;

begin

// Vraag 3.2

iRy := StrToInt(cmbRy.text);

iKol := StrToInt(cmbKol.text);

cKar := arrSpel[iRy, iKol];

Inc(iGetal);

if cKar = '#' then

begin

arrSpel[iRy, iKol] := '$';

Inc(iGevind);

end

else

begin

redV3Foutief.Lines.Add('R' + IntToStr(iRy) + ', C' +IntToStr(iKol));

end;

vertoon;

pnlV3GetalRaaiskote.Caption := IntToStr(iGetal);

if (iGevind >= 2) AND (iGetal <= 5) then

begin

btnV3\_2Speel.Enabled := false;

ShowMessage('Spel gewen');

end;

if (iGetal >= 5) AND (btnV3\_2Speel.Enabled) then

begin

btnV3\_2Speel.Enabled := false;

ShowMessage('Spel verloor');

end;

end;

// ===================================================================

// Vraag 3.3

// ===================================================================

procedure TfrmV3.btnV3\_3OnthullClick(Sender: TObject);

var

iRy, iKol: integer;

sReel: string;

begin

// Vraag 3.3

redV3Spelbord.Clear;

for iRy := 1 to Length(arrSpel) do

begin

sReel := '';

for iKol := 1 to Length(arrSpel) do

begin

sReel := sReel + arrSpel[iRy, iKol] + ' ';

end;

redV3Spelbord.Lines.Add(sReel);

end;

end;

end.