



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

NASIONALE SENIOR SERTIFIKAAT

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

FEBRUARIE/MAART 2011

MEMORANDUM

PUNTE: 120

Hierdie memorandum bestaan uit 27 bladsye.

ALGEMENE INLIGTING:

- Bladsye 2 – 12 bevat die Delphi-memoranda met moontlike oplossings vir VRAAG 1 tot 3 in programmeringskode.
- Bladsye 13 – 21 bevat die Java-memoranda met moontlike oplossings vir VRAAG 1 tot 3 in programmeringskode.
- Bladsye 22 – 27 bevat BYLAE A tot F wat 'n nasienblad vir elke vraag insluit vir kandidate wat enigeen van die twee programmeringstale gebruik.
- Kopieë van die toepaslike BYLAE behoort vir elke leerder gemaak te word om te voltooi tydens die nasiensessie.

AFDELING A: DELPHI**VRAAG 1: PROGRAMMERING EN DATABASIS (DELPHI)**

```
unit Vraag1_U;

interface

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

type
  TfrmWer = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    btnA: TButton;
    btnB: TButton;
    btnC: TButton;
    btnD: TButton;
    btnE: TButton;
    btnF: TButton;
    btnG: TButton;
    BitBtn1: TBitBtn;
    qryWer: TADOQuery;
    tblWerfAg: TDataSource;
    grdWer: TDBGrid;
    procedure btnAClick(Sender: TObject);
    procedure btnBClick(Sender: TObject);
    procedure btnCClick(Sender: TObject);
    procedure btnDClick(Sender: TObject);
    procedure btnEClick(Sender: TObject);
    procedure btnFClick(Sender: TObject);
    procedure btnGClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmWer: TfrmWer;

implementation

{$R *.dfm}
```

```

procedure TfrmWer.btnAClick(Sender: TObject);           // Vraag 1.1
begin
  qryWer.Active := False;
  qryWer.SQL.Text := 'SELECT *✓ FROM tblAgentskappe✓ ORDER BY
                                AantVorigePlasings✓ DESC'✓;
  qryWer.Active := True;
end;                                                    (4)
//=====
procedure TfrmWer.btnBClick(Sender: TObject);           // Vraag 1.2
begin
  qryWer.Active := False;
  qryWer.SQL.Text := 'SELECT Naam, Van, Salaris✓ FROM tblKliente✓ WHERE
                                Voltyds = False✓ AND Salaris < 15000✓';
  qryWer.Active := True;
end;                                                    (4)
//=====
procedure TfrmRec.btnCClick(Sender: TObject);           // Vraag 1.3
begin
  qryWer.Active := False;
  qryWer.SQL.Text := 'SELECT Count(*)✓ AS [Die aantal agentskappe wat
internasionaal werk aanbied is] ✓ FROM tblAgentskappe✓ WHERE
Internasionaal=True✓';
  qryWer.Active := True;
end;                                                    (4)
//=====
procedure TfrmRec.btnDClick(Sender: TObject);           // Vraag 1.4
begin
  qryWer.Active := False;
  qryWer.SQL.Text := 'SELECT Naam, Van, Salaris✓, format✓ (Salaris * (10 /
                                100), ".00"✓) AS [AgentKomm] ✓ FROM tblKliente✓';
  qryWer.Active := True;
end;


OF:(Salaris * 0.1, 2)


                                                    (5)
//=====
procedure TfrmRec.btnEClick(Sender: TObject);           // Vraag 1.5
begin
  qryWer.Active := False;
  qryWer.SQL.Text := 'INSERT INTO tblAgentskappe✓ VALUES✓ ("Jobs
                                Unlimited", "Wes-Kaap", False, 0) ✓✓ ';

  qryWer.ExecSQL;
  MessageDlg ( 'Rekord is suksesvol ingevoeg', mtInformation, [mbOk], 0);
  qryWer.Active := True;
end;                                                    (4)
//=====
procedure TfrmRec.btnFClick(Sender: TObject);           // Vraag 1.6
begin
  qryWer.Active := False;
  qryWer.SQL.Text := 'SELECT Naam, Van, AgentskapNaam, Provinsie ✓ FROM
tblAgentskappe, tblKliente✓ WHERE tblAgentskappe.AgentskapNaam✓ =
tblKliente.GeplaasDeur✓ AND (Provinsie = 'Wes-Kaap'✓ OR Provinsie = 'Gauteng')
'✓;
  qryWer.Active := True;
end;                                                    (6)

```

```
//=====
procedure TfrmRec.btnGClick(Sender: TObject);           // Vraag 1.7
var
    naam, datum :string;
begin
    naam := InputBox('Werwingsmaatskappy','Tik die naam van die agentskap in ',
    ''); ✓
    datum := InputBox('Werwingsmaatskappy', 'Tik die afsny-datum in', ''); ✓
    gryWer.Active := False;
    gryWer.SQL.Text := 'SELECT Naam, Van, DatumGeplaas✓ FROM tblKliente ✓WHERE
DatumGeplaas < #' + datum + '# ✓✓AND GeplaasDeur =' + naam +'" ✓✓';
    gryWer.ExecSQL;
    gryWer.Active := True;
end;                                                    (8)
//=====
end.
```

TOTAAL VRAAG 1: 35

VRAAG 2: OBJEK-GEORIËNTEERDE PROGRAMMERING (DELPHI)**unit uStadXXXX;**

interface

uses SysUtils;

//=====

// Vraag 2.1.1 (3)

type TStad = class

private ✓

stadNaam : String;	}	✓✓
graadWerk : integer;		
diplomaWerk : integer;		
salarisTotaal : real;		

public

constructor Create; overload;

constructor Create(sStad : String); overload;

procedure telbyDipWerk(rSalaris : real);

procedure telbyGrWerk(rSalaris : real);

function gemSalaris : real;

function getStadNaam : String;

function isRegteStad(rSalaris : real; sWerk : String) : boolean;

function toString : String;

end;

implementation

{ Stad }

//=====

// Vraag 2.1.2 (3)

constructor TStad.Create(sStad : String); ✓

begin

stadNaam := sStad; ✓

graadWerk := 0;

diplomaWerk := 0;

salarisTotaal := 0.00;

✓

end;

constructor TStad.Create;

begin

end;

//=====

// Vraag 2.1.3 (5)

procedure TStad.telbyGrWerk(rSalaris: real);

begin

inc(graadWerk);

salarisTotaal := salarisTotaal + rSalaris;

end;

procedure TStad.telbyDipWerk(rSalaris: real);

begin

inc(diplomaWerk);

salarisTotaal := salarisTotaal + rSalaris;

end;

✓ Haal kommentaartekens af

function TStad.gemSalaris✓: real; ✓

begin

Result := Round(salarisTotaal/(diplomaWerk + graadWerk)✓*100.0)/100.0; ✓

end;

//=====

// Vraag 2.1.4 (2)

```
function TStad.getStadNaam: String; ✓
begin
    Result := stadNaam; ✓
end;
//=====
```

// Vraag 2.1.5 (5)

```
function TStad.isRegteStad(rSalaris: real; sWerkTipe: String✓): boolean✓;
begin
    if ((UpperCase(sWerkTipe) = 'GRAAD') AND (graadWerk > diplomaWerk) AND
        (gemSalaris > rSalaris))✓ then
        begin
            Result := true;
        end
    else
        if ((UpperCase(sWerkTipe) = 'DIPLOMA') AND (diplomaWerk > graadWerk) AND
            (gemSalaris > rSalaris))✓ then
            begin
                Result := true;
            end
        else
            Result := false; ✓
    end;
//=====
```

//Vraag 2.1.6 (4)

```
function TStad.toString: String;
var
    objStr : String;
begin
    objStr := 'Stad : ' + stadNaam + #13; ✓
    objStr := objStr + 'Diploma Werk : '+intToStr(diplomaWerk)+ #9 +
        'Graad Werk : ' + ✓intToStr(graadWerk) + #13;
    objStr := objStr + 'Gemiddelde Salaris : R'✓ + floatToStr(gemSalaris); ✓
    Result := objStr;
end;
//=====
```

unit Vraag2_UXXXX;

```
interface
```

```
uses
```

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

```
type
```

```
TfrmVraag2 = class(TForm)
    redAfvoer: TRichEdit;
    MainMenu1: TMainMenu;
    mnuA: TMenuItem;
    mnuB: TMenuItem;
    Quit: TMenuItem;
    function werkKategorie(sWerk : String) : String;
    procedure QuitClick(Sender: TObject);
    procedure VertoonStadInfoClick(Sender: TObject);
    procedure CheckJobMatchClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
```

```
var
```

```
Kopiereg voorbehou
```

Blaai om asseblief

```
frmVraag2: TfrmVraag2;
```

```
implementation
```

```
{ $R *.dfm }
```

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

```
uses
```

```
    uStadXXXX;
```

```
// Vraag 2.2.1 (20)
```

```
var
```

```
    stad : TStad; ✓✓
```

```
const
```

```
    graadWerk : array[1..6] of String = ('Dokter', 'Programmeerder', 'Argitek',  
'Onderwyser', 'Regseleerde', 'Ingeneur');
```

```
    diplomaWerk : array[1..6] of String = ('Sekretaresse', 'Werktuigkundige',  
'Elektrisien', 'Skoonheidsdeskundige', 'Verpleegster', 'Loodgieter');
```

```
function TfrmVraag2.werkKategorie(sWerk: String): String;
```

```
var
```

```
    k : integer;
```

```
begin
```

```
    Result := 'Fout';
```

```
    for k := 1 to 6 do
```

```
        begin
```

```
            if (sWerk = graadWerk[k]) then
```

```
                Result := 'Graad';
```

```
            if (sWerk = diplomaWerk[k]) then
```

```
                Result := 'Diploma';
```

```
        end;
```

```
end;
```

```
procedure TfrmQuest2.QuitClick(Sender: TObject);
```

```
begin
```

```
    Application.Terminate;
```

```
end;
```

```
procedure TfrmQuest2.FormActivate(Sender: TObject);
```

```
var
```

```
    tLeer : textfile; ✓
```

```
    sStad, sWerk, sSalaris, sKategorie : String;
```

```
    iTel : integer;
```

```
    rSalaris : real
```

```
begin
```

```
    redAfvoer.Lines.Clear;
```

```
    iTel := 0; ✓
```

```
    if FileExists('Werk.txt') <> TRUE then ✓
```

```
        begin
```

```
            ShowMessage('Leer nie gevind'); ✓
```

```
            Application.Terminate;
```

```
        end
```

```
    else
```

```
        begin
```

```
            AssignFile(tLeer, 'Werk.txt'); ✓
```

```
            Reset(tLeer);
```

```
            ReadLn(tLeer, sStad); ✓
```

```
            stad := TCity.Create(sStad); ✓✓
```

```
            while NOT EOF(tLeer) do ✓
```

```
                begin
```

```
                    ReadLn(tLeer, sWerk); ✓
```

```
                    ReadLn(tLeer, sSalaris); ✓
```

```

    rSalaris := StrToFloat(sSalaris); ✓
    skategorie := werkKategorie(sWerk); ✓

    if (skategorie = 'Graad') then
        begin
            stad.telbyGrWerk(rSalaris);
        end
    else if (skategorie = 'Diploma') then
        begin
            stad.telbyDipWerk(rSalaris);
        end;
    inc(iTel); ✓
end;
CloseFile(tLeer); ✓
redAfvoer.Lines.Add('');
redAfvoer.Lines.Add(IntToStr(iTel) + ' werksgeleenthede in ' +
                    stad.getStadNaam + ' is verwerk'); ✓
end;
end;
//=====

```

// Vraag 2.2.2 (2)

```

procedure TfrmQuest2.mnuAClick(Sender: TObject);
begin
    redAfvoer.Lines.Clear;
    redAfvoer.Lines.Add(stad.toString); ✓✓
end;
//=====

```

// Vraag 2.2.3 (5)

```

procedure TfrmVraag2.mnuBClick(Sender: TObject);
var
    skategorie : String;
    rSalaris : real;
begin
    redAfvoer.Lines.Clear;
    skategorie := InputBox('Soek regte stad', 'Tik jou kwalifikasie in
                                (Graad/Diploma)', ''); ✓
    rSalaris := StrToFloat(InputBox('Soek regte stad', 'Tik jou minimum vereiste
                                salaris in ', '')); ✓

    if (stad.isRegteStad✓ (rSalaris, skategorie) ✓) then
        begin
            redAfvoer.Lines.Add(stad.getStadNaam + ' is 'n
                                goeie plek om te soek vir 'n werk');
        end
    else
        begin
            redAfvoer.Lines.Add(stad.getStadNaam + ' voldoen nie aan jou minimum
                                vereistes nie');
        end;
end;
end;
end.
//=====

```

TOTAAL VRAAG 2: 49

VRAAG 3: DELPHI-PROGRAMMERING

LET WEL: Hierdie is slegs 'n voorbeeld – leerders kan hierdie vraag beantwoord op enige manier wat hulle wil. Ken punte toe volgens beginsels wat reg toegepas is. Kan die rubriek gebruik wat voorsien is.

```

unit Vraag3_UXXXX;

interface

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

type
  TfrmVraag3 = class(TForm)
    redAfvoer: TRichEdit;
    pnlButtons: TPanel;
    btnA: TButton;
    btnB: TButton;
    BitBtn1: TBitBtn;
    procedure FormCreate(Sender: TObject);
    procedure btnAClick(Sender: TObject);
    procedure btnBClick(Sender: TObject);

    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag3: TfrmVraag3;

implementation
//=====
{$R *.dfm}
// Vraag 3.1 (12)
type
//=====
  arrTipe = array[1..20] of string;
  Alfa = Set of 'A'..'Z';
var
  arrLeerders : array[1..20] of string;
  arrBesoeikers : arrTipe; ✓
  iTel : integer;
  Konsonante : Alfa;

  procedure TfrmVraag3.FormCreate(Sender: TObject);
var
  iPlek, iPunt, K : integer;
  sVak, sNaam: string;

begin
  arrLeerders[1] := 'Susan Thompson,Wiskunde:77';
  arrLeerders[2] := 'Eric Ntumba,IT:89';
  arrLeerders[3] := 'Sean Franklin,Rekeningkunde:70';
  arrLeerders[4] := 'Mohammed Naidoo,Wiskunde:68';
  arrLeerders[5] := 'Rowan Huntley,IT:77';

```

LET WEL:

Oordrag van parameters na Sorteer subprogram is nie 'n vereiste nie. Die volgende verklaring is aanvaarbaar:

arrBesoeikers: array[1..20] of string;

```

arrLeerders[6] := 'Elizabeth Xaba,Ekonomie:77';
arrLeerders[7] := 'Sue Daniels,IT:69';
arrLeerders[8] := 'Zane Shameez,Wiskunde:90';
arrLeerders[9] := 'Mpho Anderson,Wetenskap:81';
arrLeerders[10] := 'Bryan Smith,IT:75';
arrLeerders[11] := 'Christopher Khumalo,
                    Rekeningkunde:78';
arrLeerders[12] := 'Camilla Johnson,Wetenskap:88';
arrLeerders[13] := 'Taryn Peterson,Wetenskap:70';
arrLeerders[14] := 'Jack Nelson,Wiskunde:68';
arrLeerders[15] := 'Joe Zimmerman,Wetenskap:76';
arrLeerders[16] := 'Gregory Thompson,IT:87';
arrLeerders[17] := 'Dwane Franklin,IT:89';
arrLeerders[18] := 'Sean Franklin,Rekeningkunde:70';
arrLeerders[19] := 'Mohammed Naidoo,Wiskunde:68';
arrLeerders[20] := 'Cindy Mokoena,IT:70';

```

```

iTel := 0; ✓
for K := 1 to 20 do ✓
begin
    iPlek := pos(',', arrLeerders[K]); ✓
    sNaam := copy(arrLeerders[K],1,iPlek - 1); ✓
    delete(arrLeerders[K],1,iPlek); }
    iPlek := pos(':', arrLeerders[K]); } ✓
    sVak := copy(arrLeerders[K], 1, iPlek -1); ✓
    iPunt := StrToInt(copy(arrLeerders[K], iPlek + 1, 3)); ✓
    if ((sVak = 'Wiskunde') OR (sVak = 'Wetenskap') OR (sVak = 'IT')) ✓ AND
        (iPunt >= 70) ✓ then
        begin
            inc(iTel); ✓
            arrBesoekers[iTel] := sNaam; ✓
        end;
end;
end;

```

V 3.1: 12
Om voorwaardes te toets en name toe te ken aan Besoekers skikking:
 (1) Verklaar Besoekers skikking
 (1) Inisialiseer die teller
 (1) Lus
 Haal die volgende uit die string in gegewe skikking
 (2) naam
 (2) vak
 (1) punt
 (2) if-toets voorwaardes
 if true then
 (1) inc teller
 (1) ken naam toe aan besoekers skikking

Delphi:
 Oplossing sonder Delete-stelling is aanvaarbaar om die naam, vak en punt uit te haal.
 Sien kode hieronder in kommentaarblok.

Alternatief om naam, vak en punt uit string te kry:
 Verklaar iKomma en iDubbelP as integer veranderlikes.

```

iKomma := pos(',', arrLeerders[k]);
sNaam := copy(arrLeerders[K],1,iKomma - 1);
iDubbelP := pos(':', arrLeerders[K]);
redAfvoer.Lines.Add(IntToStr(iDubbelP));
sVak := copy(arrLeerders[K], iKomma + 1, iDubbelP - iKomma -1);
redAfvoer.Lines.Add(sVak);
iPunt := StrToInt(copy(arrLeerders[K], iDubbelP + 1, 3));

```

//=====

// Vraag 3.2 (6)

```

procedure Sorteert(var arrName:arrTipe; iTel:integer);

```

```

var
    K, J :integer;
    sTemp :string;
begin
    for K := 1 to iTel -1 do ✓
        for J := k + 1 to iTel do ✓
            if arrName[K] > arrName[J] then ✓
            begin
                sTemp := arrName[K]; ✓
                arrName[K] := arrName[J]; ✓
                arrName[J] := sTemp; ✓
            end;
        end;
    end;
end;

```

Delphi:
 Oordrag van parameters is die beste oplossing maar nie 'n vereiste nie. Moenie penaliseer indien geen parameters gebruik is nie.

V 3.2: 6 punte vir sorter-metode
 (1) Buitenste lus
 (1) Binneste lus
 (1) if
 Binne die if: (3) ruil elemente reg om

Kan enige erkende sorteer-kode gebruik.

//=====

// Vraag 3.3 (4)

procedure TfrmVraag3.btnAClick(Sender: TObject);

var

K :integer;

begin

redAfvoer.Clear;

Sorteer(arrBesoekers, iTel); ✓

redAfvoer.Lines.Add('Alfabeties-Gesorteerde Lys Besoekers'); ✓

redAfvoer.Lines.Add('=====');

for K := 1 to iTel do✓

redAfvoer.Lines.Add(arrBesoekers[K]); ✓

end;

//=====

// Vraag 3.4 (14)

procedure TfrmVraag3.btnBClick(Sender: TObject);

var

K, J :integer;

sNommer, sNom :string;

begin

Randomize;

Konsonante := ['B'..'D', 'F'..'H', 'J'..'N', 'P'..'T', 'V'..'Z'];

redAfvoer.Clear;

redAfvoer.Lines.Add('Lys van Besoekers met
Studentenommers'); ✓redAfvoer.Lines.Add('=====');
redAfvoer.Lines.Add('=====');redAfvoer.Lines.Add('Naam' + #9 +
'Studentenommer'); ✓redAfvoer.Lines.Add('=====');
redAfvoer.Lines.Add('=====');

Sorteer(arrBesoekers, iTel); ✓

for K := 1 to iTel do✓

begin

sNommer := '';

for J := 1 to length(arrBesoekers) do✓

begin

if (upcase(arrBesoekers[K][J]) ✓ in
Konsonante✓) ✓ thensNommer := sNommer + upcase✓
(arrBesoekers[K][J]);

end;

sNommer := copy(sNommer, 1, 3) ;

sNommer := sNommer + ✓IntToStr(random(900) + 100); ✓✓

redAfvoer.Lines.Add(arrBesoekers[K] + #9✓ + sNommer); ✓

end;

redAfvoer.Lines.Add(' ');

end;

//=====

V 3.3: 4 punte vir Vertoon Name

(1)Sorteer skikking - roep subprogram

(1)Vertoon opskrif

(1) Lus

(1)Binne die lus: vertoon

Delphi: Alternatief:

Kan 'n versameling met klinkers in plaas van konsonante gebruik en die if-stelling met 'n NOT gebruik.

**V 3.4 : 14 punte vir
Studentenommers**(1) Sorteer Skikking
(roep subprogram)(2) Opskrif en
subopskrifte(1) Herhaal iTel keer
Binne lus:

Kry konsonante:

(1) Lus/herhaal 3x

(3) kry karakter, toets
vir konsonant

(1) hoofletters

Kry 3-syfer-getal:

(2) genereer ewekansige
getal(1) voeg konsonante en
nommer saam in eenstring voeg by string
getal bymekaar(2) vertoon naam en
studentenommer netjies

Alternatief vir binneste FOR-lus. Kan WHILE-lus gebruik:

```
iTeller := 0;
J := 1;
while (iTeller < 3) and (J <=length(arrBesoekeers)) do
begin
  if NOT(upcase(arrBesoekeers[K][J]) in ['A','E','I','O','U',' ']) then
  begin
    sNommer := sNommer + upcase(arrBesoekeers[K][J]);
    inc(iTeller);
  end;
  inc(J);
end;
```

```
procedure TfrmVraag3.FormActivate(Sender: TObject);
begin
  redAfvoer.Paragraph.TabCount := 1;
  redAfvoer.Paragraph.Tab[0] := 100;
end;

end.
```

//=====

TOTAAL VRAAG 3: 36

TOTAAL AFDELING A: 120

AFDELING B: JAVA**VRAAG 1: PROGRAMMERING EN DATABASIS (JAVA)**

```

import java.io.*;
import java.sql.*;
import javax.swing.*;
import java.util.Scanner;

public class ToetsWerwing
{
    public static void main (String[] args) throws SQLException, IOException
    {
        BufferedReader inKb = new BufferedReader (new InputStreamReader
(System.in));
        Werwing DB = new Werwing();
        System.out.println();
        char keuse = ' ';
        do
        {
            System.out.println("      KEUSELYS");
            System.out.println();
            System.out.println("      Opsie A");
            System.out.println("      Opsie B");
            System.out.println("      Opsie C");
            System.out.println("      Opsie D");
            System.out.println("      Opsie E");
            System.out.println("      Opsie F");
            System.out.println("      Opsie G");
            System.out.println();
            System.out.println("      V - VERLAAT");
            System.out.println(" ");
            System.out.print("      Jou Keuse? ");
            keuse = inKb.readLine().toUpperCase().charAt(0);
            System.out.println(" ");
            String sql = "";
            switch(keuse)
            {
                case 'A':
                    //Vraag 1.1
                    sql = "SELECT *✓FROM tblAgentskappe✓ORDER BY AantVorigePlasings✓
DESC✓";
                    DB.A(sql);
                    break;
                //=====
                case 'B':
                    // Vraag 1.2
                    sql = "SELECT Naam, Van, Salaris✓ FROM tblKliente✓ WHERE
Voltyds = False✓ AND Salaris < 15000✓";
                    DB.B(sql);
                    break;
                //=====
                case 'C':
                    // Vraag 1.3
                    sql = "SELECT Count(*)✓ AS [Tel] ✓FROM tblAgentskappe ✓
WHERE Internasionaal = True"✓;
                    DB.C(sql);
                    break;
            }
        }
    }
}

```

```

//=====
        case 'D': // Vraag 1.4
        {
            sql = "SELECT Naam, Van, Salaris✓, round✓(Salaris * (10 / 100), 2)✓
                AS [AgentKomm] ✓ FROM tblKliente✓";

            DB.D(sql);
            break;
        }
        OF:(Salaris * 0.1, 2)
        (5)
//=====
        case 'E': // Vraag 1.5
        {
            Scanner kb = new Scanner (System.in);
            System.out.println();

            sql = "INSERT INTO tblAgentskappe ✓
                VALUES ✓('Jobs Unlimited', 'Wes-Kaap', False, 0) ✓✓";
            DB.E(sql);

            break;
        }
        (4)
//=====
        case 'F': // Vraag 1.6
        {
            sql = "SELECT Naam, Van, AgentskapNaam, Provinsie ✓FROM
                tblAgentskappe, tblKliente ✓
                WHERE tblAgentskappe.AgentskapNaam✓
                = tblKliente.GeplaasDeur ✓AND (Provinsie = 'Wes-Kaap'✓
                OR Provinsie = 'Gauteng') ✓";

            DB.F(sql);

            break;
        }
        (6)
//=====
        case 'G': // Vraag 1.7
        {
            Scanner kb = new Scanner (System.in);
            System.out.println("Tik die naam van die agentskap in ");
            String agentskapNaam = kb.nextLine();✓
            System.out.println("Tik die afsny-datum in ");
            String datum = kb.next();✓

            sql = "SELECT Naam, Van, DatumGeplaas ✓FROM tblKliente✓
                WHERE DatumGeplaas < #" + datum + "#✓✓AND
                GeplaasDeur ='"+ agentskapNaam + "'✓✓";

            DB.G(sql);
            break
        }
        (8)
//=====
    }
    }while (keuse != 'V');

    DB.disconnect();
    System.out.println("Klaar");
}
}
//=====

```

TOTAAL VRAAG 1: 35

VRAAG 2: OBJEK-GEORIËNTEERDE PROGRAMMERING (JAVA)**Stad.java**

```

public class Stad
{
    // Vraag 2.1.1 (3)
    private ✓ String stadNaam;
    private int graadWerk;
    private int diplomaWerk;
    private double salarisTotaal; } ✓✓

    public Stad()
    {
    }
}

//=====
// Vraag 2.1.2 (3)
public Stad(String naam) ✓
{
    stadNaam = naam; ✓

    graadWerk = 0;
    diplomaWerk = 0;
    salarisTotaal = 0; } ✓
}

//=====
// Vraag 2.1.3 (5)
public void telbyDipWerk(double salaris)
{
    diplomaWerk ++;
    salarisTotaal = salarisTotaal + salaris;
}

public void telbyGrWerk(double salaris)
{
    graadWerk ++;
    salarisTotaal = salarisTotaal + salaris;
}

public double ✓ gemSalaris() ✓
{
    double salaris = Math.round(salarisTotaal / (graadWerk + diplomaWerk)*
                                ✓ 100) / 100.0; ✓
    return salaris;
}

//=====
// Vraag 2.1.4 (2)
public String getStadNaam () ✓
{
    return stadNaam; ✓
}

//=====
// Vraag 2.1.5 (5)
public boolean ✓ isRegteStad(double sal, String kategorie ✓)
{
    if (kategorie.equalsIgnoreCase("graad") & graadWerk > diplomaWerk &
        gemSalaris() > sal) ✓
    {
        return true;
    }
}

```

✓ Haal kommentaartekens af

```

        else if (kategorie.equalsIgnoreCase("diploma") & diplomaWerk >
                                graadWerk & gemSalaris()
                                > sal) ✓
        {
            return true;
        }
        else
        {
            return false; ✓
        }
    }
}
//=====
//Vraag 2.1.6 (4)

public String toString()
{
    String objStr = "";

    objStr = objStr + "Stad : " + stadNaam + "\n";✓
    objStr = objStr + "Diploma Werk : " + diplomaWerk + "\t"
                + "Graad Werk : " + graadWerk + "\n";✓
    objStr = objStr + "Gemiddelde Salaris : R"✓ + gemSalaris() + "\n";✓

    return objStr;
}
}

```

```

//=====
ToetsStad.java

```

```

import javax.swing.*;
import java.io.*;
import java.util.*;

public class ToetsStad
{
    public static String graadWerk [] = {"Dokter", "Programmeerder", "Argitek",
    "Onderwyser", "Regseleerde", "Ingeneur"};
    public static String diplomaWerk[] = {"Sekretaresse", "Werktuigkundige",
    "Elektrisien", "Skoonheidsdeskundige", "Verpleegster", "Loodgieter"};

    public static String werkKategorie(String werktitel)
    {
        String werktipe = "Error";

        for(int i = 0; i < 6; i++)
        {
            if (graadWerk[i].equals(werktitel))
            {
                werktipe = "Graad";
            }
            else if (diplomaWerk[i].equals(werktitel))
            {
                werktipe = "Diploma";
            }
        }

        return werktipe;
    }
}

```

//=====

// Vraag 2.2.1 (20)

```

public static void main(String args[]) throws Exception
{
    Stad stad = new Stad();✓✓

    File leer = new File("Werk.txt"); ✓

    if (!leer.exists())✓    {
        System.out.println("Leer nie gevind"); ✓
        System.exit(0);
    }
    else
    {
        Scanner in = new Scanner(leer); ✓
        String sStad = in.nextLine();✓

        stad = new Stad(sStad); ✓✓

        int tel = 0; ✓

        while (in.hasNextLine())✓
        {
            String werk = in.nextLine();✓
            double salaris = Double.parseDouble(in.nextLine());✓✓

            String werktipe = werkKategorie(werk); ✓

            if (werktipe.equals("Graad"))
            {
                stad.telbyGrWerk(salaris);
            }
            else if (werktipe.equals("Diploma"))
            {
                city.telbyDipWerk(salaris);
            }
        } ✓✓

        tel++; ✓
    }
    System.out.println(tel + " werksgeleenthede van " +
                        stad.getStadNaam() + " is verwerk");✓
    System.out.println();
    in.close(); ✓
}

```

//=====

```

BufferedReader inKb = new BufferedReader (new InputStreamReader (System.in));
char keuse = ' ';
while (keuse != 'V')
{
    System.out.println();
    System.out.println("      KEUSELYS");
    System.out.println(" ");
    System.out.println("      A - Opsie A");
    System.out.println("      B - Opsie B");
    System.out.println(" ");
    System.out.println("      V - VERLAAT");
    System.out.println(" ");
    System.out.print("      Jou Keuse? :");
}

```

```

        keuse = inKb.readLine().toUpperCase().charAt(0);
        switch (keuse)
//=====
{
    // Vraag 2.2.2                (2)
        case 'A':
        {
            System.out.println();
            System.out.println(stad.toString());✓✓
            break;
        }
//=====
    // Vraag 2.2.3                (5)
        case 'B':
        {
            System.out.print("Tik jou kwalifikasie in (Graad/Diploma) : ");
            String kwalifikasie = inKb.readLine();✓
            System.out.print("Tik jou minimum vereiste salaris in : ");
            double salaris = Double.parseDouble(inKb.readLine());✓

            if (stad.isRegteStad✓(salaris, kwalifikasie✓))
            {
                System.out.println(stad.getStadNaam() + " is 'n goeie
                    plek om te soek vir 'n werk");
            }
            else
            {
                System.out.println(stad.getStadNaam() + " voldoen nie
                    aan jou minimum vereistes nie");
            }
            break;
        }

        case 'V':
        {
            System.exit(0);
        } // case

    } // switch

} // while

} // main

} // class

```

TOTAAL VRAAG 2: 49

VRAAG 3: JAVA PROGRAMMERING

LET WEL: Hierdie is slegs 'n voorbeeld – leerders kan hierdie vraag beantwoord op enige manier wat hulle wil. Ken punte toe volgens beginsels wat reg toegepas is. Kan die rubriek gebruik wat voorsien is.

ToetsBesoekers.java

```
import java.util.Scanner;
import java.io.*;
import javax.swing.*;
//=====
// Vraag 3.1 (12)
public class ToetsBesoekers
{
    static String[] arrLeerders = new String[20];
    static String[] arrBesoekers = new String[20]; ✓

    public static void main(String[] args) throws Exception
    {
        arrLeerders [0] = "Susan Thompson,Wiskunde:77";
        arrLeerders [1] = "Eric Ntumba,IT:89";
        arrLeerders [2] = "Sean Franklin,Rekeningkunde:70";
        arrLeerders [3] = "Mohammed Naidoo,Wiskunde:68";
        arrLeerders [4] = "Rowan Huntley,IT:77";
        arrLeerders [5] = "Elizabeth Xaba,Ekonomie:77";
        arrLeerders [6] = "Sue Daniels,IT:69";
        arrLeerders[7] = "Zane Shameez,Wiskunde:90";
        arrLeerders [8] = "Mpho Anderson,Wetenskap:81";
        arrLeerders [9] = "Bryan Smith,IT:75";
        arrLeerders [10]= "Christopher Khumalo,
                           Rekeningkunde:78";
        arrLeerders [11] = "Camilla Johnson,Wetenskap:88";
        arrLeerders [12] = "Taryn Peterson,Wetenskap:70";
        arrLeerders [13] = "Jack Nelson,Wiskunde:68";
        arrLeerders [14] = "Joe Zimmerman,Wetenskap:76";
        arrLeerders [15] = "Gregory Thompson,IT:87";
        arrLeerders [16] = "Dwane Franklin,IT:89";
        arrLeerders [17] = "Sean Franklin,Rekeningkunde:70";
        arrLeerders [18] = "Mohammed Naidoo,Wiskunde:68";
        arrLeerders [19] = "Cindy Mokoena,IT:70";

        int tel = 0; ✓
        for (int k = 0; k < 20; k++)✓
        {
            int komma = arrLeerders[k].indexOf(','); ✓
            String naam = arrLeerders[k].substring(0,komma); ✓

            int dubbelpunt = arrLeerders[k].indexOf(':'); ✓
            String vak = arrLeerders[k].substring(komma + 1,dubbelpunt ); ✓

            int punt = Integer.parseInt(arrLeerders[k].substring(dubbelpunt+1)); ✓
            if(((vak.equals("Wiskunde")) || (vak.equals("Wetenskap")) ||
                (vak.equals("IT")))) ✓&& (punt >= 70)) ✓
            {
                arrBesoekers[tel] = naam; ✓
                tel++;✓
            }
        }
    }
}
```

V 3.1: 12 punte vir
toets van leerders en
toeken aan
arrBesoekers:

(1) Verklaar
Besoekers-skikking
(1) Inisialiseer die
teller
(1) Lus
Haal die volgende uit
die string in gegewe
skikking
(2) naam
(2) vak
(1) punt
(2) if-toets
voorwaardes
if true then
(1) inc teller
(1) ken naam toe
aan Besoekers-
skikking

```

Scanner invoer = new Scanner(System.in);
char keuse = ' ';
while (keuse != 'V')
{
    System.out.println();
    System.out.println("          KEUSELYS");
    System.out.println(" ");
    System.out.println("          A - Opsie A");
    System.out.println("          B - Opsie B");
    System.out.println(" ");
    System.out.println("          V - VERLAAT");
    System.out.println(" ");
    System.out.print("          Jou Keuse? :");

    keuse = invoer.nextLine().toUpperCase().charAt(0);
    switch (keuse)
    {
        case 'A':
            {
                //=====

```

// Vraag 3.3 (14)

```

        sorteer(arrBesoekers, tel); ✓
        System.out.println(" ");
        System.out.println("Alfabeties-Gesorteerde Lys van Besoekers"); ✓
        System.out.println("=====");
        for (int k = 0; k < tel; k++) ✓
        {
            System.out.println(arrBesoekers [k]); ✓
        }
        break;
    }
}
//=====

```

// Vraag 3.4 (14)

```

        case 'B':
        {
            sorteer(arrBesoekers, tel); ✓
            System.out.println(" ");
            System.out.println("Lys van Besoekers met
                               Studentenommers"); ✓
            System.out.println("=====
                               =====");
            System.out.printf("%-25s%-10s\n", "Naam",
                               "Studentenommer"); ✓
            System.out.println("=====
                               =====");

```

V 3.3: 4 punte vir die Vertoon van die Name

- (1)Sorteer skikking
- (1)Vertoon opskrif
- (1) Lus
- (1)Binne die lus: vertoon

V 3.4: 14 punte vir studentenommers

- (1) Sorteer Skikking
- (roep subprogram)
- (2) Opskrif en subopskrifte
- (1) Herhaal iTel keer
- Binne lus:
- Kry konsonante:
- (1) Lus/herhaal 3x
- (3) kry karakter, toets vir konsonant
- (1) hoofletters
- Kry 3-syfer-getal:
- (2) genereer ewekansige getal
- (1) voeg konsonante en nommer saam
- (2) vertoon die naam en die studentenommer netjies

```

String nommer;
for (int k = 0; k < tel; k++) ✓
{
    nommer = "";

    for (int j = 0; j < arrBesoekers [k].length(); j++) ✓ // OF while
    {
        char letter = arrBesoekers[k].toUpperCase().charAt(j); ✓
        if (!(letter == 'A') && !(letter == 'E') && !(letter == 'I') &&
            !(letter == 'O') && !(letter == 'U') && !(letter == ' ')) ✓
            nommer = nommer + letter; ✓
    }
}

```

```

    }
    nommer = nommer.substring(0,3) ;
    nommer = nommer + ✓ (int) (Math.random () * (900) + 100); ✓✓

    System.out.printf("%-25s%-10s\n"✓, arrBesoekeers [k],nommer✓);
  }
  break;
}

case 'V':
{
  System.exit(0);
} // case

} // switch

} // while
}
//=====
// Vraag 3.2 (6)
public static void sorteer(String [] arrName, int tel)
{
  for (int k = 0; k < tel -1; k++)✓
    for (int j = k + 1; j < tel; j++)✓
    {
      if (arrName[k].compareTo(arrName[j]) > 0) ✓
      {
        String temp = arrName[k]; ✓
        arrName[k] = arrName[j]; ✓
        arrName[j] = temp; ✓
      } // end if
    } // end for
  } // end metode
}

//=====

```

V 3.2: 6 punte vir sorteer-metode

(1) Buitenste lus

(1) Binneste lus

(1) if

Binne die if:

(3) ruil elemente reg om

Kan enige erkende sorteer-kode gebruik.

TOTAAL VRAAG 3: 36**TOTAAL AFDELING B: 120**

BYLAAG A**VRAAG 1: DELPHI – PROGRAMMERING EN DATABASIS**

SENTRUMNOMMER:		EKSAMENNOMMER:.....	
VRAAG 1: DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
1.1	'SELECT *✓ FROM tblAgentskappe✓ ORDER BY AantVorigePlasings✓ DESC'✓	4	
1.2	'SELECT Naam, Van, Salaris✓ FROM tblKliente✓ WHERE Voltyds = False✓ AND Salaris < 15000✓'	4	
1.3	'SELECT Count(*)✓ AS [Die aantal agentskappe wat internasionaal werk aanbied is] ✓ FROM tblAgentskappe✓ WHERE Internasionaal = True✓'	4	
1.4	'SELECT Naam,Van,Salaris✓,format✓ (Salaris * (10 / 100),".00"✓)AS [AgentKomm] ✓ FROM tblKliente✓ OF (salary * 0.1 , 2)	5	
1.5	'INSERT INTO tblAgentskappe✓ VALUES✓ ("Jobs Unlimited","Wes-Kaap",False,0) ✓✓ '	4	
1.6	'SELECT Naam,Van,AgentskapNaam,Provinsie ✓FROM tblAgentskappe, tblKliente✓ WHERE tblAgentskappe.AgentskapNaam✓ = tblKliente.GeplaasDeur✓ AND (Provinsie = 'Wes-Kaap'✓ OR Provinsie = 'Gauteng') '✓	6	
1.7	Toevoer naam, ✓ toevoer datum✓ 'SELECT Naam, Van, DatumGeplaas✓ FROM tblKliente ✓WHERE DatumGeplaas < #' + datum + '#' ✓✓AND GeplaasDeur =""+ naam +'"" ✓✓	8	
	TOTAAL:	35	

BYLAAG B

VRAAG 2: DELPHI – OBJEK-GEORIENTEERDE PROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:.....	
VRAAG 2: DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
2.1			
2.1.1	Definieer attribute vir TStad: vier privaat (1) velde, met korrekte datatipes (1) en name (1)	3	
2.1.2	Konstruktor: parameter vir naam (1) ken parameterwaarde toe aan stadNaam-veld (1) inisialiseer ander velde na nul (1)	3	
2.1.3	gemSalaris metode: verwyder kommentaar van telbyDipWerk en telbyGrWerk metodes (1). Metode met die naam gemSalaris (1) wat real terugstuur (1) bevat korrekte berekening (1) en afgerond tot twee desimale plekke (1)	5	
2.1.4	getStadNaam metode: Regte naam en stuur String terug (1), Stuur regte veld terug (1)	2	
2.1.5	isRegteStad metode: Ontvang een real en een String parameter (1) stuur Boolean waarde terug (1) regte voorwaardes om regte stad te bepaal (2) en stuur true/false reg terug (1)	5	
2.1.6	toString metode: voeg tabs (1) en nuwe reels (1) by bestaande kode en voeg gemiddelde salaris by (1) met opskrif en tab (1)	4	
2.2			
2.2.1	• Verklaar 'n enkele TStad objek (2) • Verklaar 'n nuwe veranderlike vir die lêer (1) • Verklaar 'n teller en initialiseer dit na 0 (1) • As die lêer nie bestaan nie (1) vertoon boodskap en staak program (1) • As die lêer bestaan doen assignfile (1) • Lees die eerste lyn uit leer buite die lus na die stadNaam veranderlike (1) • Roep die konstruktor Create van TStad (1) Gebruik die naam van die stad as 'n parameter (1) • Lus (1) om deur lêer te lees • Lees lyne (reëls) uit lêer – een vir werk (1) en een vir salaris (1) salaris moet na 'n real omgeskakel word (1), • Roep werkKategorie subprogram (1) en vergelyk die resultaat (1) roep die regte metode (telbyDipWerk / telbyGrWerk) in TStad (1), • Vermeerder die teller (1) en vertoon die resultaat (1) • Maak die lêer toe buite lus (1)	20	
2.2.2	Roep toString-metode van TStad-objek (2)	2	
2.2.3	• Lees kwalifikasie (1) en salaris (1) in van sleutelbord • Roep isRegteStad-metode in die TStad-klas (1) met die regte twee arguments (kategorie en salaris) (1) • Vertoon die regte afvoer gebaseer op die Boole-waarde wat teruggestuur word (1)	5	
	TOTAAL:	49	

BYLAAG C**VRAAG 3: DELPHI–PROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:.....	
VRAAG 3: DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
3.1	12 punte vir toets van leerders en toeken van name aan Besoekers skikking (1) Verklaar Besoekers skikking (1) Inisialiseer die teller (1) Lus om deur leerders skikking te lees Haal die volgende uit die string in gegewe skikking (2) naam (2) vak (1) punt (2) if- toets voorwaardes if true then (1) inc teller (1) ken naam toe aan besoeker skikking	12	
3.2	6 punte vir subprogram (1) Buitenste lus (1) Binneste lus (1) if Binne die if: (3) ruil elemente reg om	6	
3.3	4 punte vir Vertoon Name (1) Sorteër skikking (Roep subprogram) (1) Vertoon opskrif (1) Lus (1) Binne die lus: vertoon	4	
3.4	14 punte vir Studentennommers (1) Sorteër Skikking (Roep subprogram) (2) Opskrif en subopskrifte (1) Herhaal iTel keer Binne lus (vir elke leerder): Kry konsonante: (1) Lus / herhaal 3x (3) kry karakter, toets vir konsonant (1) hoofletters Kry 3-syfer getal: (2) genereer ewekansige getal (1) voeg konsonante en nommer saam in een string (2) vertoon die naam en die studentenommer netjies	14	
	TOTAAL:	36	

BYLAAG D**VRAAG 1: JAVA – PROGRAMMERING EN DATABASIS**

SENTRUMNOMMER:		EKSAMENNUMMER:.....	
VRAAG 1: JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
1.1	SELECT *✓FROM tblAgentskappe✓ORDER BY AantVorigePlasings✓	4	
1.2	SELECT Naam, Van, Salaris✓ FROM tblKliente✓ WHERE Voltyds = False✓ AND Salaris < 15000✓	4	
1.3	SELECT Count(*)✓ AS [Tel] ✓FROM tblAgentskappe ✓ WHERE Internasionaal = True✓	4	
1.4	SELECT Naam, Van, Salaris✓, round✓(Salaris * (10 / 100), 2)✓AS [AgentKomm] ✓ FROM tblKliente✓ OF (Salaris * 0.1, 2)	5	
1.5	INSERT INTO tblAgentskappe ✓ VALUES ✓('Jobs Unlimited', 'Wes-Kaap', False, 0) ✓✓	4	
1.6	SELECT Naam, Van, AgentskapNaam, Provinsie ✓FROM tblAgentskappe, tblKliente ✓WHERE tblAgentskappe.AgentskapNaam✓ = tblKliente.GeplaasDeur ✓AND (Provinsie = 'Wes-Kaap'✓OR Provinsie = 'Gauteng') ✓	6	
1.7	Toevoer Agentnaam ✓ Toevoer Datum ✓ SELECT Naam, Van, DatumGeplaas✓ FROM tblKliente✓ WHERE DatumGeplaas < #" + datum + "#✓✓AND GeplaasDeur = '" + agentskapNaam + "'✓✓	8	
	TOTAAL:	35	

BYLAAG E

VRAAG 2: JAVA – OBJEK-GEORIËNTEDE PROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:.....	
VRAAG 2: JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
2.1			
2.1.1	Definieer attribute vir Stad: vier privaat (1) velde, met korrekte datatypes (1) en name (1)	3	
2.1.2	Konstruktor: parameter vir naam (1) ken parameterwaarde toe aan stadNaam-veld (1) inisialiseer ander velde na nul (1)	3	
2.1.3	gemSalaris metode: verwyder kommentaar van telbyDipWerk en telbyGrWerk metodes (1). Metode met die naam gemSalaris (1) wat real terugstuur (1) bevat korrekte berekening (1) en afgerond tot twee desimale plekke (1)	5	
2.1.4	getStadNaam metode: Regte naam en stuur String terug (1), Stuur regte veld terug (1)	2	
2.1.5	isRegteStad metode: Een double en een String parameter (1) stuur Boolean terug (1) regte voorwaardes om regte stad te bepaal (2) en stuur true/false reg terug (1)	5	
2.1.6	toString metode: voeg tabs (1) en nuwe lyne (reels) (1) by bestaande kode en voeg gemiddelde salaris by (1) met opskrif en tab (1)	4	
2.2			
2.2.1	• Verklaar 'n enkele Stad objek (2) • Verklaar 'n nuwe veranderlike vir die lêer objek (new File(1)) • Verklaar 'n teller en inisialiseer dit na 0 (1) • As die lêer nie bestaan nie (1) vertoon boodskap en staak program (1) • As die lêer bestaan skep nuwe Scanner-objek om uit lêer te lees (1) • Lees die eerste lyn uit leer buite die lus na die stadNaam veranderlike (1) • Roep die konstruktor van Stad (1) Gebruik die naam van die stad as 'n parameter (1) • Lus (1) om deur lêer te lees • Lees lyne (reels) uit lêer – een vir werk (1) en een vir salaris (1) salaris moet 'n real omgeskakel word (1), • Roep werkKategorie subprogram (1) en vergelyk die resultaat (1) roep die regte metode (telbyDipWerk / telbyGrWerk) in TStad (1), • Vermeerde die teller (1) en vertoon die resultaat (1) • Maak die lêer toe buite lus (1)	20	
2.2.2	Roep toString metode van Stad objek (2)	2	
2.2.3	• Lees kwalifikasie (1) en salaris (1) in van sleutelbord • Roep isRegteStad-metode in die Stad-klas (1) met die regte twee arguments (kategorie en salaris) (1) • Vertoon die regte afvoer gebaseer op die Boolean waarde wat teruggestuur word (1)	5	
	TOTAAL:	49	

BYLAAG F**VRAAG 3: JAVA-PROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:.....	
VRAAG 3: JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
3.1	12 punte vir toets van leerders en toeken van name aan Besoekers skikking (1) Verklaar Besoekers skikking (1) Inisialiseer die teller (1) Lus om deur leerders skikking te lees Haal die volgende uit die string in gegewe skikking (2) naam (2) vak (1) punt (2) if- toets voorwaardes if true then (1) inc teller (1) ken naam toe aan besoeker skikking	12	
3.2	6 punte vir subprogram (1) Buitenste lus (1) Binneste lus (1) if Binne die if: (3) ruil elemente reg om	6	
3.3	4 punte vir Vertoon Name (1) Sorteër skikking (Roep metode) (1) Vertoon opskrif (1) Lus (1) Binne die lus: vertoon	4	
3.4	14 punte vir Studentennommers (1) Sorteër Skikking (Roep metode) (2) Opskrif en subopskrifte (1) Herhaal iTel keer Binne lus (vir elke leerder): Kry konsonante: (1) Lus / herhaal 3x (3) kry karakter, toets vir konsonant (1) hoofletters Kry 3-syfer getal: (2) genereer ewekansige getal (1) voeg konsonante en nommer saam in een string (2) vertoon die naam en die studentenommer netjies	14	
	TOTAAL:	36	