



education

Department:
Education
REPUBLIC OF SOUTH AFRICA

NASIONALE SENIOR SERTIFIKAAT

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

FEBRUARIE/MAART 2010

MEMORANDUM

PUNTE: 120

Die memorandum bestaan uit 25 bladsye en 7 bylae.

Algemene inligting:

- Bladsye 2 – 11 bevat die Delphi memoranda met moontlike oplossings vir VRAE 1 tot 3 in programmeringskode.
- Bladsye 12 – 24 bevat die Java memoranda met moontlike oplossings vir VRAE 1 tot 3 in programmeringskode.
- Bladsye 25 – 32 bevat Bylae A tot G wat 'n dekblad insluit sowel as nasienblad vir elke vraag vir kandidate wat enigeen van die twee programmeringstale gebruik.
- Kopieë van die 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
```

```
TfrmVetV = class(TForm)
```

```
  pnlGroot: TPanel;
```

```
  Panel2: TPanel;
```

```
  btnVerskil: TButton;
```

```
  btnGeskiedenis: TButton;
```

```
  btnLede: TButton;
```

```
  btnLaag: TButton;
```

```
  btnPasAan: TButton;
```

```
  BitBtn1: TBitBtn;
```

```
  btnAgter: TButton;
```

```
  qryVetV: TADOQuery;
```

```
  tblVetV: TDataSource;
```

```
  grdVetV: TDBGrid;
```

```
  procedure btnVerskilClick(Sender: TObject);
```

```
  procedure btnGeskiedenisClick(Sender: TObject);
```

```
  procedure btnLaagClick(Sender: TObject);
```

```
  procedure btnAgterClick(Sender: TObject);
```

```
  procedure btnPasAanClick(Sender: TObject);
```

```
  procedure btnLedeClick(Sender: TObject);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```
end;
```

```
var
```

```
  frmVetV: TfrmVetV;
```

```
implementation
```

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

```
procedure TfrmVetV.btnLedeClick(Sender: TObject); //Vraag 1.1
```

```
begin
```

```
  qryVetV.Active := False;
```

```
  qryVetV.SQL.Text := 'SELECT *✓FROM tblMense✓ ORDER BY✓ InskryfDatum Desc✓';
```

```
  qryVetV.Active := True;
```

```
end;
```

(5)

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

```

procedure TfrmVetV.btnVerskilClick(Sender: TObject); //Vraag 1.2
begin
  qryVetV.Active := False;
  qryVetV.SQL.Text := 'SELECT PersoonID, Naam, B_Gewig, D_Gewig, ✓' +
    'ROUND(✓(B_Gewig - D_Gewig)✓, 1) ✓ AS KgsOmTeVerloor ✓FROM tblMense✓';
  qryVetV.Open;
end; (6)
=====
procedure TfrmVetV.btnAgterClick(Sender: TObject); //Vraag 1.3
begin
  qryVetV.Active := False;
  qryVetV.SQL.Text := 'DELETE ✓FROM tblMense ✓WHERE✓ Balans > 400 ✓and ✓NOT(Naam
= "Uriel Knight")'; ✓
  qryVetV.ExecSQL;
  ShowMessage('Agterstallige rekeninge uitgegee.');
```

end; (6)

```

=====
procedure TfrmVetV.btnLaagClick(Sender: TObject); //Vraag 1.4
begin
  qryVetV.Active := False;
  qryVetV.SQL.Text := 'SELECT PersoonID, MIN(Gewig) ✓ AS MinGewig ✓FROM ' +
    'tblWeegin ✓GROUP BY✓ PersoonID✓';
  qryVetV.ExecSQL;
  qryVetV.Open;
end; (6)
=====
procedure TfrmVetV.btnGeskiedenisClick(Sender: TObject); //Vraag 1.5
var
  id : string;
begin
  id := InputBox('Individuele inweeg-geskiedenis ', 'Tik die ID in ', '1'); ✓
  qryVetV.Active := False;
  qryVetV.SQL.Text := 'SELECT tblMense.PersoonID, ✓ Naam, D_Gewig, Gewig, ' +
    'WeegDatum ✓FROM tblMense, ✓ tblWeegin✓ WHERE ' +
    'tblMense.PersoonID ✓= tblWeegin.PersoonID✓ AND✓ ' +
    'tblMense.PersoonID ✓= ' + id; ✓

  qryVetV.ExecSQL;
  qryVetV.Open;
end; (10)
=====
procedure TfrmVetV.btnPasAanClick(Sender: TObject); //Vraag 1.6
begin
  qryVetV.Active := False;
  qryVetV.SQL.Text := 'UPDATE tblWeegin ✓SET ✓Gewig ✓= Gewig✓ * 1.1 ✓WHERE ' +
    'MONTH(WeegDatum) = 5✓';
  qryVetV.ExecSQL;
  qryVetV.SQL.Text := 'SELECT * FROM tblWeegin';
  qryVetV.Active := True;
end; (7)
=====
end.
```

VRAAG 2: OBJEK-GEORIENTEERDE PROGRAMMERING

```

unit EteXXXX;

interface
uses
  SysUtils;
Type
  TEte = class✓
  private✓
    fDag      :String; } ✓
    fEtenstyd :String; }
    fVette    :integer; } ✓
    fKoolh    :integer; }
    fProteien :integer; }
  public✓
    constructor Create(sDag, sEte : String; iVette,iProt,iKoolh :
                                integer); ✓

    function getDag : String;
    function geenVette : boolean;
    function berekenPunte : integer;
    function toString:String;
  end;

```

```

implementation

```

// V 2.1.2. (4 ÷ 2 = 2)

```

constructor TEte.Create✓ (sDag, sEte : string; iVette,iProt,iKoolh :
                                integer); ✓

begin
  fDag := sDag;
  fEtenstyd := sEte;
  fVette := iVette;
  fProteien := iProt;
  fKoolh := iKoolh;
end;

```

// V 2.1.3. (4 ÷ 2 = 2)

```

function TEte.getDag✓: String; ✓
begin
  result ✓:= fDag; ✓
end;

```

// V 2.1.4. (4 ÷ 2 = 2)

```

function TEte.geenVette: boolean✓;
begin
  if fVette = 0 ✓then
    result := true✓
  else
    result := false✓;
end;

```

OF

```

result := false✓;
if fVette = 0✓ then
  result := true;✓

```

// V 2.1.5.

(6 ÷ 2 = 3)

```
function TEte.berekenPunte: integer; ✓
var
  tot : integer;
begin
  tot := (fVette * 4) + (fProteien * 2) + (fKoolh * 2); ✓✓
  if geenVette then
    tot := tot - 2; ✓
  if fVette > 2 then
    tot := tot + 10; ✓
  result := tot; ✓
end;
```

// V 2.1.6.

(6 ÷ 2 = 3)

```
function TEte.toString:String✓;
begin
  result := fDag✓ + #9✓ + intToStr✓ (fVette) + #9 + intToStr(fProteien)
+ #9 + intToStr(fKoolh) + #9 + intToStr (berekenPunte) ✓ // roep metode
+ #9 + fEtenstyd; ✓ // vir al die velde
end;
```

end.

```
//=====
unit Vraag2XXXX_U;
```

interface

uses

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

type

```
TfrmVraag2 = class(TForm)
  MainMenu1: TMainMenu;
  DagVerslag: TMenuItem;
  GeenVette: TMenuItem;
  Verlaat: TMenuItem;
  redAfvoer: TRichEdit;
  BesteSwakste: TMenuItem;
  procedure VerlaatClick(Sender: TObject);
  procedure FormActivate(Sender: TObject);
  procedure DagVerslagClick(Sender: TObject);
  procedure GeenVetteClick(Sender: TObject);
  procedure BesteSwaksteClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

var

```
frmVraag2: TfrmVraag2;
```

implementation

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

// V 2.2.1.

(26 ÷ 2 = 13)

```

uses EteXXXX; ✓
var
  arrEtes :array[1..100] of TEte✓;
  iTel    :integer; ✓

{$R *.dfm}

procedure TfrmVraag2.FormActivate(Sender: TObject);
var
  TLeer :TextFile; ✓
  sEenlyn, sDag, sEtenstyd :String;
  iVette, iProteiene, iKoolhidrate, iHash :integer;

begin
  if fileExists('Etes.txt') ✓ <> true then
    begin
      ShowMessage('Die teksleer ''Etes.txt'' bestaan nie'); ✓
      Exit; ✓
    end;
  AssignFile(TLeer, 'Etes.txt'); ✓
  Reset(TLeer); ✓
  iTel := 0; ✓
  While not eof(TLeer) ✓ do
    begin
      inc(iTel); ✓ // MOET initialiseer voor lus en inkrementeer binne lus
      readln(TLeer, sEenlyn); ✓
      iHash := pos('#', sEenlyn); ✓
      sDag := copy(sEenlyn, 1, iHash -1); ✓
      delete(sEenlyn, 1, iHash); ✓

      iHash := pos('#', sEenlyn);
      sEtenstyd := copy(sEenlyn, 1, iHash -1); ✓
      delete(sEenlyn, 1, iHash); ✓

      iHash := pos('#', sEenlyn);
      iVette := StrToInt✓ (copy(sEenlyn, 1, iHash -1)); ✓
      delete(sEenlyn, 1, iHash);

      iHash := pos('#', sEenlyn);
      iProteiene := StrToInt(copy(sEenlyn, 1, iHash -1)); ✓
      delete(sEenlyn, 1, iHash);

      iKoolhidrate := StrToInt(sEenlyn); ✓

      arrEtes[iTel] ✓ := TEte.Create✓ (sDag, sEtenstyd, iVette,
                                      iProteiene, iKoolhidrate); ✓
    end;
  CloseFile(TLeer); ✓

end;
//=====

```

// V 2.2.2.

(14 ÷ 2 = 7)

```

procedure TfrmVraag2.DagVerslagClick(Sender: TObject);
var
  K      : integer;
  iTotaal : integer;
  sDagVanWeek : String;
begin
  sDagVanWeek := InputBox('Dag', 'Tik die dag is (e.g. Maa, Din, ens)', 'Son');
  iTotaal := 0;
  redAfvoer.Clear;
  redAfvoer.Lines.add('Inligting oor Etes vir ' + upperCase(sDagVanWeek));
  redAfvoer.Lines.add('');
  redAfvoer.Lines.add('Dag' + #9 + 'Vette' + #9 + 'Prot' + #9 +
    'Koolh' + #9 + 'Punte' + #9 + 'Ete');
  redAfvoer.Lines.add('');
  For K := 1 to iTel do
  begin
    if arrEtes[K].getDag = sDagVanWeek then
    begin
      redAfvoer.Lines.add(arrEtes[K].toString);
      iTotaal := iTotaal + arrEtes[K].berekenPunte;
    end;
  end;
  redAfvoer.Lines.Add(' ');
  if iTotaal = 0 then
    ShowMessage('Geen Etes gevind vir ' + sDagVanWeek)
  else
    redAfvoer.Lines.add('Die totale hoeveelheid punte is ' +
      IntToStr(iTotaal));
    if iTotaal > 50 then
      redAfvoer.Lines.add('Perke Oorskry')
    else
      redAfvoer.Lines.add('Binne Perke');
  end;
end;

```

OF

```

if iTotaal <= 50 then
  redAfvoer.Lines.add('Binne
  Perke')
else
  redAfvoer.Lines.add('Perke
  Oorskry');

```

//=====

// V 2.2.3.

(6 ÷ 2 = 3)

```

procedure TfrmVraag2.GeenVetteClick(Sender: TObject);
var
  K : integer;
begin
  redAfvoer.Clear;
  redAfvoer.Lines.add('Inligting oor Etes met Geen Vette');
  redAfvoer.Lines.add('');
  redAfvoer.Lines.add('Dag' + #9 + 'Vette' + #9 + 'Prot' + #9 +
    'Koolh' + #9 + 'Punte' + #9 + 'Ete');
  redAfvoer.Lines.add('');
  For K := 1 to iTel do
  begin
    if arrEtes[K].GeenVette then
    begin
      redAfvoer.Lines.add(arrEtes[K].toString);
    end;
  end;
end;

```

//=====

// V 2.2.4.

(12 ÷ 2 = 6)

```

procedure TfrmVraag2.BesteSwaksteClick(Sender: TObject);
var
    K, iHoogste, iLaagste : integer;
    sHoogste, sLaagste    : String;
begin
    redAfvoer.Clear;
    redAfvoer.Lines.add('Etes met die Meeste en Minste Punte');
    redAfvoer.Lines.add('');
    redAfvoer.Lines.add(#9 + #9+'Dag'+ #9 + 'Vette' + #9 +'Prot'+ #9 +
        'Koolh' + #9 + 'Punte' + #9 + 'Ete');
    redAfvoer.Lines.add('');
    iHoogste := arrEtes[1].berekenPunte;
    iLaagste := arrEtes[1].berekenPunte;
    sHoogste := arrEtes[1].ToString;
    sLaagste := arrEtes[1].ToString;

    For K := 2 to iTel do
    begin
        if arrEtes[K].berekenPunte > iHoogste then
        begin
            iHoogste := arrEtes[K].berekenPunte;
            sHoogste := arrEtes[K].ToString;
        end;

        if arrEtes[K].berekenPunte < iLaagste then
        begin
            iLaagste := arrEtes[K].berekenPunte;
            sLaagste := arrEtes[K].ToString;
        end;

        redAfvoer.Lines.add('Hoogste' + #9 + sHoogste);
        redAfvoer.Lines.add('Laagste' + #9 + sLaagste);
    end;

    procedure TfrmVraag2.VerlaatClick(Sender: TObject);
    begin
        Close;
    end;

end.

```

✓

✓✓ Initialiseer veranderlikes voor lus

Begin moet saam met end wees op regte plek vir 1 punt

✓ Herhaal met < vir Laagste Moenie weer vir vorige foute penaliseer nie

VRAAG 3: DELPHI PROGRAMMERING

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

```
unit SMSKompXXXX_U;

interface

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

type
  TfrmSMSKomp = class(TForm)
    mnuSMSKomp: TMainMenu;
    mnuVerwerking: TMenuItem;
    mnuLys: TMenuItem;
    mnuWenners: TMenuItem;
    redAfvoer: TRichEdit;
    mnuStoor: TMenuItem;
    mnuVerlaat: TMenuItem;

    procedure mnuLysClick(Sender: TObject);
    procedure mnuWennersClick(Sender: TObject);
    procedure QuitClick(Sender: TObject);
    procedure mnuVerlaatClick(Sender: TObject);
    procedure FormCreate(Sender: TObject); private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmSMSKomp: TfrmSMSKomp;
  arrInskrywings : array[1..20] of string;
  arrKorrek : array[1..20] of string; ✓
  arrWenners: array[1..3] of string;
  iTeller, iTel :integer;

implementation

{$R *.dfm}
procedure TfrmSMSKomp.FormCreate(Sender: TObject);
begin
  arrInskrywings[1] := '082 345 4571:Oefening';
  arrInskrywings[2] := '082543 2341:Oefening';
  arrInskrywings[3] := '082 234 1241:OEFENING';
  arrInskrywings[4] := '0821239876:Eet';
  arrInskrywings[5] := '083123 6123:Slaap';
  arrInskrywings[6] := '083 452 4353:OEFENING';
  arrInskrywings[7] := '0831009844:Slaap';
  arrInskrywings[8] := '083 104 2333:Oefening';
  arrInskrywings[9] := '076 239 6966:Slaap';
  arrInskrywings[10] := '076986 4533:EET';
  arrInskrywings[11] := '076 365 4272:Oefening';
  arrInskrywings[12] := '076563 2642:Oefening';
  arrInskrywings[13] := '084 884 1244:OEFENING';
  arrInskrywings[14] := '0841239867:Slaap';
  arrInskrywings[15] := '084123 6444:Oefening';
```

```

arrInskrywings[16] := '084 1156 434:Oefening';
arrInskrywings[17] := '079 1229 844:Eet';
arrInskrywings[18] := '079 456 2331:Oefening';
arrInskrywings[19] := '079 239 7971:OEFENING';
arrInskrywings[20] := '079986 6622:EET';
iTeller := 20;
end;
//=====
function verwyderSpasies(sNommer:string):string; ✓
var
    K           :integer;
    sNuweNommer:string;
begin
    sNuweNommer := ''; ✓
    for K := 1 to length(sNommer) do ✓
        begin
            if (sNommer[K] <> ' ') ✓ then
                sNuweNommer := sNuweNommer + sNommer[K]; ✓
            end;
            Result := sNuweNommer; ✓
        end;
    end;
//=====
procedure TfrmSMSKomp.mnuLysClick(Sender: TObject);
var
    K, iPos      :integer;
    sNommer, sAntwoord:string;
begin
    iTel := 0;
    for K := 1 to iTeller do ✓
        begin
            iPos := pos(':', arrInskrywings[K]); ✓
            sNommer := copy(arrInskrywings[K], 1, iPos - 1); ✓
            sNommer := verwyderSpasies(sNommer); ✓
            delete(arrInskrywings[K], 1, iPos); ✓
            sAntwoord := arrInskrywings[K]; ✓
            if (uppercase(sAntwoord) = 'OEFENING') ✓ then
                begin
                    iTel := iTel + 1; ✓ // moet iTel initialiseer
                    arrKorrek[iTel] := sNommer; ✓
                end;
            end;
        end;
    redAfvoer.Clear;
    redAfvoer.Lines.Add('Selfoonnommers van
                        moontlike weners');
    for K := 1 to iTel do
        redAfvoer.Lines.Add(arrKorrek[K]);
    end;
end;
//=====

```

V 3.1 : 6 punte vir subprogram
 (1) Funksie definisie
 (1) Initialiseer nuwe string
 (1) Stap met lus deur string
 Binne lus:
 (1) toets of nie spasie dan
 (1) voeg karakter by nuwe string
 Na die lus:
 (1) stuur nuwe string terug

V 3.1 : 13 punte
 (1) Stap met lus deur
 gegewe skikking
 (2) Kry selnummer uit
 string
 (1) Roep funksie om
 spasies te verwyder
 (2) Kry antwoord uit
 string
 (1) As die antwoord reg is
 (1) ignoreer hoofletters
 Verhoed duplikate
 (3 punte):
EEN VAN
 (1) Verklaar nuwe
 skikking
 (1) gebruik teller vir
 nuwe skikking reg
 (1) voeg nommer in nuwe
 skikking in
OF
 Gebruik gegewe
 skikking(1) deur te toets
 vir duplikate in die
 huidige skikking(2)
 Vertoon inligting: opskrif
 en lus(1), vertoon(1)

```
procedure TfrmSMSKomp.mnuWennersClick(Sender: TObject);
```

```
var
```

```
    K, iRandomGetal :integer;
```

```
begin
```

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

```
    redAfvoer.Lines.Add('Lys van wenners'); ✓
```

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

```
    Randomize;
```

```
    for K := 1 to 3 do✓
```

```
        begin
```

```
            repeat
```

```
                iRandomGetal := Random(iTel) + 1; ✓
```

```
            until arrKorrek[iRandomGetal] <> ''; ✓
```

```
            if (arrKorrek[iRandomGetal] <> '') then✓
```

```
                begin
```

```
                    arrWenners[K] ✓:= arrKorrek[iRandomGetal]; ✓
```

```
                    redAfvoer.Lines.Add('Winner #' + IntToStr(K)
```

```
                        + ' : '✓ +arrWenners[K]); ✓
```

```
                    arrKorrek[iRandomGetal] := ''; ✓
```

```
                end;
```

```
        end;
```

```
    end;
```

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

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

```
var
```

```
    TLeer :TextFile;
```

```
    K      :integer;
```

```
begin
```

```
    AssignFile(TLeer, 'Wenners.txt'); ✓
```

```
    Rewrite(TLeer); ✓
```

```
    writeln(TLeer, 'Lys van wenners'); ✓
```

```
    for K := 1 to 3 do✓
```

```
        begin
```

```
            writeln(TLeer, 'Wenner # ' + IntToStr(K) + ' : ' + arrWenners[K]); ✓✓
```

```
        end;
```

```
    CloseFile(TLeer); ✓
```

```
end;
```

```
procedure TfrmSMSKomp.mnuVerlaatClick(Sender: TObject);
```

```
begin
```

```
    Close;
```

```
end;
```

V 3.2 : 10 punte

Gebruik nuwe skikking / of 3 veranderlikes

(1) Herhaal 3 keer / gebruik 3 veranderlikes

(2) Genereer random getal

(4) Enige metode om wenners wat reeds gekies is nie weer te kies nie

MOONTLIKE OPLOSSING:

Gebruik lus om nommer te kies totdat nommer gevind wat nog nie gekies is nie(1) bv. nommer wat leeg is in ou skikking

Indien nie gevind, plaas nommer(1) in nuwe skikking / veranderlike(1)

Verwyder / of dui op 'n manier aan in ou skikking dat nommer reeds gekies is(1)

(1) Vertoon opskrif en

(1) wenners

(1) in korrekte formaat

V 3.3: 7 punte

(1) gebruik regte lêernaam

(1) maak lêer oop vir skryf

(1) skryf opskrif na lêer

(2) skryf 3 keer na lêer

(wenners)

(1) in korrekte formaat

(1) maak lêer toe

TOTAAL AFDELING A: 120

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

```

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

public class ToetsVetV
{
    public static void main (String[] args) throws
SQLException,IOException
    {
        BufferedReader inKb = new BufferedReader (new InputStreamReader
                                                    (System.in));

        VetVegters DB = new VetVegters();
        System.out.println("\f");

        char keuse = ' ';
        do
        {
            System.out.println("          KEUSELYS");
            System.out.println();
            System.out.println("    A - Alle Lede");
            System.out.println("    B - Gewigsverskille");
            System.out.println("    C - Agterstallige Rekeninge");
            System.out.println("    D - Laagste Gewig");
            System.out.println("    E - Weeggeskiedenis");
            System.out.println("    F - Pas Gewig Aan");
            System.out.println();
            System.out.println("    V - VERLAAT PROGRAM");
            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 tblMense✓ ORDER BY✓ InskryfDatum
                                                                    DESC"✓;
                        DB.vertoonAlmal(sql);
                        break;
                    }
                    //=====
                    case 'B':
                        // Vraag 1.2
                        {
                            sql = "SELECT PersoonID, Naam, B_Gewig, D_Gewig, ✓ ROUND✓
                                (B_Gewig - D_Gewig✓, 1) ✓ AS KgsOmTeVerloor✓ FROM
                                                                    tblMense✓";
                            DB.berekenGewigVerskil(sql);
                            break;
                        }
                    //=====

```

```

        case 'C':
            //Vraag 1.3
            {
                sql = "DELETE✓ FROM tblMense✓ WHERE✓ Balans > 400✓ AND✓
Naam                <>'Uriel Knight'✓";
                DB.verwyderUitstaande(sql);
                break;
            }
            //===== (6)
        case 'D':
            //Vraag 1.4
            {
                sql = "SELECT MIN(Gewig) ✓ AS MinGewig✓, PersoonID✓ FROM
                tblWeegin✓ GROUP BY✓
PersoonID✓";
                DB.bepaalLaagste(sql);
                break;
            }
            //===== (6)
        case 'E':
            // Vraag 1.5
            {
                System.out.print("Tik die PersoonID van die lid in: ");
                String id = inKb.readLine();✓
                sql = "SELECT tblMense.PersoonID✓, Naam, D_Gewig,
WeegDatum✓,
                Gewig FROM tblMense✓, tblWeegin✓ WHERE
                tblMense.PersoonID✓ = tblWeegin.PersoonID✓ AND✓
                tblMense.PersoonID✓ = " + id✓;
                DB.vertoonGeskiedenis(sql);
                break;
            }
            //===== (10)
        case 'F':
            //Vraag 1.6
            {
                sql = "UPDATE tblWeegin✓ SET✓ Gewig✓ = Gewig✓ * 1.1✓ WHERE
                MONTH(WeegDatum) ✓ = 5✓";
                DB.pasGewigAan(sql);
                break;
            }
            //===== (7)
    }while (keuse != 'V');

    DB.disconnect();
    System.out.println("Gedoen");
}

}

import java.sql.*;
import java.io.*;
import javax.swing.JOptionPane;

public class VetVegters
{
    Connection conn;

```

```

public VetVegters ()
{
    //laai die driver
    try
    {
        Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");
        System.out.println ("Drywer suksesvol gelaai");
    }
    catch (ClassNotFoundException c)
    {
        System.out.println ("Nie instaat om die databasis drywer
            te laai nie");
    }

    //koppel met die databasis
    try
    {

//conn = DriverManager.getConnection ("jdbc:odbc:VetVegtersDB.mdb");

//System.out.print("Tik die presiese plek in waar jou databasis
    gestoor is (BYVOORBEELD - C:/TOETS/VetVegtersDB.mdb)");
// Vir toevoer van die sleutelbord af...
// String filename = inKb.nextLine();
// OR BufferedReader inKb = new BufferedReader (new InputStreamReader
    (System.in));
// String filename = inKb.readLine();
// OF kodeer die JChooser komponent om die lêer op te soek

    String leernaam = "VetVegtersDB.mdb";

    String databasis = "jdbc:odbc:Driver={Microsoft Access Driver
        (*.mdb)};DBQ=";
    databasis += leernaam.trim () + ";DriverID=22;READONLY=true}";
    conn = DriverManager.getConnection (databasis, "", "");

    System.out.println ("Konneksie met die VetVegters databasis is
        suksesvol bewerkstellig");

    }
    catch (Exception e)
    {
        System.out.println ("Kan nie met die databasis koppel nie");
    }
} //end konneksie

public void vertoonAlmal (String sql) throws SQLException
{
    System.out.println();
    Statement stmt = conn.createStatement ();

    ResultSet rs = stmt.executeQuery (sql);
    System.out.printf ("%10s%-18s%-8s%-8s%-15s%-
11s", "PersoonID", "Naam", "B_Gewig", "D_Gewig", "InskryfDatum", "Balans");
    System.out.println();

```

```

System.out.println("=====");

while (rs.next ())
{
    String sId = rs.getString ("PersoonID");
    String sNaam = rs.getString ("Naam");
    String sBegin = rs.getString ("B_Gewig");
    String sDoel = rs.getString ("D_Gewig");
    String sInskryfDatum = rs.getString("InskryfDatum");
    sInskryfDatum = sInskryfDatum.substring(0,10);
    double dBal = Double.parseDouble(rs.getString("Balans"));
    dBal = Math.round(dBal * 100) / 100.0;
    System.out.printf("%-10s%-18s%-8s%-10s%-13s  %-11s",sId
,sNaam,sBegin, sDoel,sInskryfDatum, dBal);
    System.out.println();
}

System.out.println(" ");
rs.close();
stmt.close ();
}

public void berekenGewigVerskil (String sql)throws SQLException
{
    Statement stmt = conn.createStatement ();

    ResultSet rs = stmt.executeQuery (sql);
    System.out.printf("%-10s%-18s%-8s%-8s%-
10s","PersoonID","Naam","B_Gewig", "D_Gewig", "KgsOmTeVerloor");
    System.out.println();

System.out.println("=====");
while (rs.next ())
{
    String sId = rs.getString ("PersoonID");
    String sNaam = rs.getString ("Naam");
    String sBegin = rs.getString ("B_Gewig");
    String sDoel = rs.getString ("D_Gewig");
    String sKg = rs.getString ("KgsOmTeVerloor");

    //int kgs = (int)Double.parseDouble(sKg);

    System.out.printf("%-10s%-18s%-8s%-10s%-8s",sId
,sNaam,sBegin, sDoel, sKg);
    System.out.println();
}
System.out.println(" ");
rs.close();
stmt.close ();
}

public void verwyderUitstaande (String sql)throws SQLException
{
    System.out.println();
    Statement stmt = conn.createStatement ();

    int rye = stmt.executeUpdate (sql);

```

```

        System.out.println(rye + " rekords is uitgegee");

        stmt.close ();
    }

    public void bepaalLaagste(String sql)throws SQLException
    {
        System.out.println();
        Statement stmt = conn.createStatement ();

        ResultSet rs = stmt.executeQuery (sql);
        System.out.printf("%-10s%-14s", "PersoonID", "MinGewig");
        System.out.println();
        System.out.printf("=====");
        System.out.println();
        while (rs.next ())
        {
            String sID = rs.getString ("PersoonID");
            double dGewig =
Double.parseDouble(rs.getString("MinGewig"));
            int Gewig = (int)Math.round(dGewig);

            System.out.printf("%-10s%-14s", sID, Gewig);
            System.out.println();
        }
        System.out.println(" ");
        stmt.close ();
    }

    public void vertoonGeskiedenis(String sql)throws SQLException
    {
        System.out.println();
        Statement stmt = conn.createStatement ();

        ResultSet rs = stmt.executeQuery (sql);
        System.out.printf("%-10s%-18s%-8s%-8s%-14s", "PersoonID", "Naam",
"D_Gewig", "Gewig", "WeegDatum");
        System.out.println();

        System.out.println("=====");
        while (rs.next ())
        {

            String sId = rs.getString ("PersoonID");
            String sNaam = rs.getString ("Naam");
            String sDoel = rs.getString ("D_Gewig");
            double dGewig = Double.parseDouble(rs.getString("Gewig"));
            int Gewig = (int)Math.round(dGewig);

            String sWeegDatum = rs.getString("WeegDatum");
            sWeegDatum = sWeegDatum.substring(0,10);

            System.out.printf("%-10s%-18s%-8s%-8s%-14s", sId, sNaam,
sDoel,

                                                                    Gewig, sWeegDatum)
                                                                    ;

            System.out.println();
        }
    }

```



```
        System.out.println(" ");
        stmt.close ();
    }
    public void pasGewigAan(String sql)throws SQLException
    {
        System.out.println();
        Statement stmt = conn.createStatement ();

        int rye = stmt.executeUpdate (sql);

        System.out.println(rye + " rekords opgedateer");

        sql = "SELECT * FROM tblWeegin";

        ResultSet rs = stmt.executeQuery (sql);
        System.out.printf("%-10s%-15s%-14s", "PersoonID", "WeegDatum",
                                                                    "Gewig");

        System.out.println();
        System.out.printf("=====");
        System.out.println();
        while (rs.next ())
        {
            String sID = rs.getString ("PersoonID");
            String sWeegDatum = rs.getString("WeegDatum");
            sWeegDatum = sWeegDatum.substring(0,10);
            String sGewig = rs.getString ("Gewig");

            System.out.printf("%-10s%-15s%-14s",sID, sWeegDatum,
sGewig);

                System.out.println();
            }
        System.out.println(" ");
        stmt.close ();
    }

    public void disconnect () throws SQLException
    {
        conn.close ();
    }
}
```

VRAAG 2: OBJEK-GEORIENTEERDE PROGRAMMERING**EteXXXX.java****// V 2.1.1. (6 ÷ 2 = 3)**

```

public class✓ EteXXXX           ✓ Klasnaam dieselfde as lêernaam
{                                ✓ Metodes public
    private✓ String dag; ✓
    private String ete;
    private int vette; ✓
    private int proteiene;
    private int koolhidrate;

```

// V 2.1.2. (4 ÷ 2 = 2)

```

public EteXXXX(✓String Dag, String eTyd, int vet, int prot, int koolh) ✓
{
    dag = Dag;
    ete = eTyd;
    vette = vet;
    proteiene = prot;
    koolhidrate = koolh;
}

```

} ✓✓

// V 2.1.3. (4 ÷ 2 = 2)

```

public String✓ getDag()✓
{
    return✓ dag;✓
}

```

// V 2.1.4. (4 ÷ 2 = 2)

```

public boolean✓ geenVette()
{
    if (vette == 0) ✓
    {
        return true✓;
    }
    else
    {
        return false✓;
    }
}

```

OF

```

boolean result = false✓;
if (vette == 0)✓
    result = true;✓
return result;

```

// V 2.1.5. (6 ÷ 2 = 3)

```

public int✓ berekenPunte()
{
    int punte = vette * 4 + proteiene * 2 + koolh * 2✓✓;

    if (vette > 2)
    {
        return (punte + 10); ✓
    }
    else if (geenVette())
    {
        return (punte - 2); ✓
    }
    else
    {
        return punte; ✓
    }
}

```

// V 2.1.6. (6 ÷ 2 = 3)

```

public String✓ toString()
{
    return dag✓ + "\t\t"✓ + vette ✓ + "\t\t" + proteiene + "\t\t\t" +
        koolh + "\t\t" + berekenPunte()✓ // roep methode
        + "\t\t\t" + ete; ✓ // al die velde
}
} // einde van klas

```

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

ToetsEte.java

```

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

public class ToetsEte
{
    public static void main(String args[]) throws Exception
    {

```

// V 2.2.1. (26 ÷ 2 = 13)

```

    EteXXXX[]✓ arrEte = new EteXXXX[100]; ✓
    int teller = 0; ✓

    File f ✓= new File("Etes.txt");✓

    if (f.exists())✓ // met else vir 1 punt
    {

        Scanner sc = new Scanner(f); ✓

        while ✓ (sc.hasNextLine())✓
        {
            String eenlyn = sc.nextLine();✓

            String tokens[]✓= line.split✓ ("#");✓

```

```

String dag = tokens[0]; ✓
String ete = tokens[1]; ✓
int vette = Integer.parseInt(✓ (tokens[2])); ✓
int proteiene = Integer.parseInt(tokens[3]); ✓
int koolh = Integer.parseInt (tokens[4]); ✓

arrEte[teller✓] = new✓ EteXXXX(dag, ete, vette,
proteiene, koolh); ✓
teller++;✓
}

sc.close(); ✓
}
else
{
    System.out.println("Leer bestaan nie");✓
    System.exit(0); ✓
}

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 - Daaglikse Verslag");
    System.out.println("    B - Etes sonder Vette");
    System.out.println("    C - Beste en Swakste Etes");
    System.out.println(" ");
    System.out.println("    V - VERLAAT PROGRAM");
    System.out.println(" ");
    System.out.print("    Jou keuse? :");

    keuse = inKb.readLine().toUpperCase().charAt(0);

    switch (keuse)
    {

```

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

// V 2.2.2.

(14 ÷ 2 = 7)

```

case 'A':
{
    System.out.println();

    System.out.print("Tik 'n dag in : " );
    String dag = inKb.readLine().toUpperCase();✓
    int totPunte = 0; ✓

    System.out.println();
    System.out.println("Inligting oor Etes vir " +
dag); ✓
    System.out.println();

    System.out.println("Dag\t\tVet\t\t
Proteiene\t\tKoolh\tPunte\t\tEte");✓

    System.out.println();

```

```

for(int i = 0; i < teller; i++)✓
{
    if(arrEte[i] ✓.getDag()✓.toUpperCase().
        equals(dag) ✓)
    {
        totPunte = totPunte ✓ +
        arrEte[i].berekenPunte()✓;

        System.out.println(arrEte[i]) ✓;
    }
}

```

```

System.out.println();
System.out.println("Daaglikse Totaal : " + totPunte); ✓

```

OF

```

if totPunte <= 50
{
    System.out.println("Binne
    Perke");
}
else
{System.out.println("Perk
    Oorskrei");
}

```

```

if (totPunte > 50) ✓
{
    System.out.println("Perk Oorskrei");
}
else
{
    System.out.println("Binne Perke");
}
break;

```

} ✓

```

//=====

```

// V 2.2.3.**(6 ÷ 2 = 3)**

```

case 'B':
{
    System.out.println();
    System.out.println("Inligting oor etes met Geen
    Vette");✓

    System.out.println();

    System.out.println("Dag\t\tVet\t\tProteiene\t\tKoolh\tPunte\t\tEte");✓
    System.out.println();

    for(int i = 0; i < teller; i++)✓
    {
        if (arrEte[i] ✓.geenVette())✓
        {
            System.out.println(arrEte[i]) ✓;
        }
    }

    System.out.println();
    break;
}
//=====

```

// V 2.2.4.

(12 ÷ 2 = 6)

```

        case 'C':
        {
            System.out.println();
            System.out.println("Etes met die Meeste en Minste
                                Punte");
            System.out.println();
        } ✓

        System.out.println("\t\tDag\t\tVet\t\tProteiene\t\tKoolh\tPunte\t\tEte");
        System.out.println();

        int maksPunte = arrEte[0];
        int maksDag = 0;
        // of String maksDag = arrEte[0].toString();
        int minPunte = arrEte[0];
        int minDag = 0;
        // of String minDag = arrEte[0].toString();
    } ✓✓ Initialiseer veranderlikes voor lus

    for(int i = 1; i < teller; i++) ✓
    {
        int punte = arrEte[i].berekenPunte(); ✓

        if (punte < minPunte) ✓
        { ✓
            // Hierdie hakie saam met
            minPunte = punte; ✓ // afsluithakie vir 1 punt
            minDag = i; ✓
            // of minDag = arrEte[i].toString();

        }

        if (punte > maksPunte)
        {
            maksPunte = punte;
            maksDag = i;
            // of maksDag = arrEte[i].toString();
        }
    } ✓ Herhaal met < vir laagste

    } Moenie weer hier penaliseer vir vorige foute

    System.out.print("Hoogste\t");
    System.out.println(arrEte[maksDag]); ✓
    // OF System.out.println(maksDag.toString());

    System.out.print("Laagste\t");
    System.out.println(arrEte[minDag]); ✓
    // OF System.out.println(minDag.toString());
    System.out.println();
    break;
}
case 'V':
{
    System.exit(0);
} // case
} // switch
} // while

} // main

} // class

```

VRAAG 3: JAVA PROGRAMMERING**ToetsSMSKompetisieXXXX.java**

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

public class ToetsSMSKompetisieXXXX
{
    static String[] arrInskrywings = new String[20];
    static String[] arrKorrek = new String[20];
    static String[] arrWenners = new String[3];
    static int teller = 0;
    static String antwoord = "OEFENING";

    public static void main(String[] args) throws Exception
    {

        arrInskrywings[0] = "082 345 4571:Oefening";
        arrInskrywings[1] = "082543 2341:Oefening";
        arrInskrywings[2] = "082 234 1241:OEFENING";
        arrInskrywings[3] = "0821239876:Eet";
        arrInskrywings[4] = "083123 6123:Slaap";
        arrInskrywings[5] = "083 452 4353:OEFENING";
        arrInskrywings[6] = "0831009844:Slaap";
        arrInskrywings[7] = "083 104 2333:Oefening";
        arrInskrywings[8] = "076 239 6966:Slaap";
        arrInskrywings[9] = "076986 4533:EET";
        arrInskrywings[10] = "076 365 4272:Oefening";
        arrInskrywings[11] = "076563 2642:Oefening";
        arrInskrywings[12] = "084 884 1244:OEFENING";
        arrInskrywings[13] = "0841239867:Slaap";
        arrInskrywings[14] = "084123 6444:Oefening";
        arrInskrywings[15] = "084 1156 434:Oefening";
        arrInskrywings[16] = "079 1229 844:Eet";
        arrInskrywings[17] = "079 456 2331:Oefening";
        arrInskrywings[18] = "079 239 7971:OEFENING";
        arrInskrywings[19] = "079986 6622:EET";

        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 - Onttrek Moontlike Wenners");
            System.out.println("      B - Kies en Vertoon Wenners");
            System.out.println("      C - Stoor Wenners");
            System.out.println(" ");
            System.out.println("      V - VERLAAT PROGRAM");
            System.out.println(" ");
            System.out.print("      Jou keuse? :");

            keuse = inKb.readLine().toUpperCase().charAt(0);

            switch (keuse)
            {
                case 'A':
                {
```

```

        onttrekKorrek();
        break;
    }

    case 'B':
    {
        kiesWenners();
        break;
    }

    case 'C':
    {
        skryfWenners();
        break;
    }

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

    } // switch

    } // while
}

//=====
public static String verwyderSpasies(String s) ✓
{
    String sonderSpasies = ""; ✓

    for(int i = 0; i < s.length(); i++) ✓
    {

        if (s.charAt(i) != ' ') ✓
        {

            sonderSpasies = sonderSpasies + s.charAt(i); ✓
        }

    }

    return sonderSpasies; ✓
}
//=====
public static void onttrekKorrek()
{
    for(int i = 0; i < arrInskrywings.length; i++) ✓
    {
        String selnom = arrInskrywings[i].substring(0,
            arrInskrywings[i].indexOf(":")); ✓✓
        selnom = verwyderSpasies(selnom); ✓

        String ans = arrInskrywings[i].substring
            (arrInskrywings[i].indexOf(":") + 1); ✓✓

        if (ans.equalsIgnoreCase(antwoord)) ✓
        {
            arrKorrek[teller] = selnom; ✓
            teller++; ✓
        }
    }
}

```

V 3.1 : 6 punte vir metode

- (1) Definisie van metode
- (1) Initialiseer nuwe string
- (1) Stap met lus deur nommer
- Binne lus:
 - (1) toets of nie spasie dan
 - (1) voeg karakter by nuwe string
- Na die lus:
 - (1) stuur nuwe string terug

V 3.1 : 13 punte

- (1) Stap met lus deur gegewe skikking
- (2) Kry selnommer uit string
- (1) Roep funksie om spasie te verwyder
- (2) Kry antwoord uit string
- (1) As die antwoord reg is
- (1) ignoreer hoofletters

Verhoed duplikate
(3 punte):

EEN VAN:

Verklaar nuwe skikking(1)
gebruik teller vir nuwe skikking reg(1)
voeg nommer in nuwe skikking in (1)

OF

Toets vir duplikate in die huidige skikking(1)
Verwyder duplicate uit gegewe skikking(2)

Vertoon inligting: opskrif en lus(1), vertoon(1)


```

        System.out.println("Selfoonnommers van
                           moontlike weners");

        for(int i = 0; i < teller; i++)✓
        {
            System.out.println(arrKorrek[i]); ✓
        }
    }
}

//=====
public static void kiesWenners() throws Exception
{
    for(int i = 0; i < arrWenners.length; i++)✓
    {
        int randGetal = (int) (Math.random()✓ *
                               teller); ✓

        arrWenners[i] = arrKorrek[randGetal]; ✓

        for(int j = randGetal; j < teller - 1; j++)✓
        {
            arrKorrek[j] = arrKorrek[j+1]; ✓
        }

        teller--;✓
    }
    System.out.println("\n");
    System.out.println("Lys van weners");✓
    for(int i = 0; i < arrWenners.length; i++)✓
    {
        System.out.println("Wenner #" + (i+1) + " : " + arrWenners[i]); ✓
    }
}

//=====
public static void skryfWenners() throws Exception
{
    PrintWriter✓ fTeksleer = new PrintWriter(new File("Wenners.txt"));✓
    fTeksleer.println("Lys van Weners"); ✓
    for(int i = 0; i < arrWenners.length; i++)✓
    {
        fTeksleer.println✓ ("Wenner #" + (i+1) + " : " + arrWenners[i]);
    }

    fTeksleer.close();✓
}

}

//=====

```

V 3.2 : 10 punte

Kan nuwe skikking skep / 3

Veranderlikes gebruik

(1) Herhaal 3 keer / ken
waardes toe aan 3

veranderlikes

(2) Genereer random getal

(4) Enige metode om weners
wat reeds gekies is nie
weer te kies nie

MOONTLIKE OPLOSSING:

Gebruik lus om te toets of
nommer in ou skikking is(1),
Indien nie, plaas in nuwe
skikking(1), verwyder
nommer(1) uit ou skikking(1)

(1) Vertoon opskrif en

(1) weners

(1) in korrekte formaat

V 3.3: 7 punte

(1) gebruik regte lêernaam

(1) maak lêer oop vir skryf

(1) skryf opskrif na lêer

(2) skryf (weners) na lêer

- herhaal 3 keer

(1) in korrekte formaat

(1) maak lêer toe

EINDE VAN AFDELING B: JAVA**TOTAAL AFDELING B: 120**

BYLAAG A**GRAAD 12 MAART 2010****INLIGTINGSTEGNOLOGIE V1****DEKBLAD****Provinsie:** _____**Sentrumnommer:** _____**Eksamennummer:** _____**Programmeringstaal (onderstreep die taal wat gebruik is):**
DELPHI / JAVA

TOTALE PUNTE PER VRAAG		
VRAAG	PUNT UIT	LEERDER SE PUNT
1	40	
2	44	
3	36	
GROOTTOTAAL	120	

BYLAAG B – Maart 2010**VRAAG 1: DELPHI - PROGRAMMERING EN DATABASIS**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG 1: DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS PUNTE	LEERDER SE PUNTE
1.1	SELECT✓ *✓ FROM tblMense✓ ORDER BY✓ InskryfDatum DESC✓	5	
1.2	SELECT PersoonID, Naam, B_Gewig, D_Gewig, ✓ ROUND✓(B_Gewig – D_Gewig✓, 1) ✓ AS KgsOmTeVerloor✓ FROM tblMense✓	6	
1.3	DELETE✓ FROM tblMense✓ WHERE✓ Balans > 400✓ AND✓ Naam <>'Uriel Knight'✓	6	
1.4	SELECT MIN(Gewig) ✓ AS MinGewig✓, PersoonID✓ FROM tblWeegin✓ GROUP BY✓ PersoonID✓	6	
1.5	Sleutelbord-toevoer van gebruiker ✓ SELECT tblMense.PersoonID✓, Naam, D_Gewig, WeegDatum✓, Gewig FROM tblMense✓, tblWeegin✓ WHERE tblMense.PersoonID✓ = tblWeegin.PersoonID✓ AND✓ tblMense.PersoonID✓ = " + id✓	10	
1.6	UPDATE tblWeegin✓ SET✓ Gewig✓ = Gewig✓ * 1.1✓ WHERE MONTH(WeegDatum) ✓ = 5✓	7	
	TOTAAL:	40	

BYLAAG C – Maart 2010

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 klas (1) TEteXXXX , vyf privaat (1) velde , 2 strings(1), 3 integers(1), Metodes public (1) Konstruktor (1) (6 ÷ 2 = 3)	3	
2.1.2	Konstruktor: Parameters in regte volgorde (1), regte tipes (1) toegeken aan velde (2) (4 ÷ 2 = 2)	2	
2.1.3	getDag (1) metode, regte tipe teruggestuur (1) regte veld (1) teruggestuur (1) (4 ÷ 2 = 2)	2	
2.1.4	geenVette metode: stuur Boolean terug(1) Toets of vette vir 0(1) stuur true terug(1) anders stuur false terug(1) (4 ÷ 2 = 2)	2	
2.1.5	berekenPunte: stuur heelgetal terug (1), regte berekeninge (2) tel 10 punte by vir meer as 2 vette (1) trek 2 punte af vir geen vette (1) stuur total terug (1) (6 ÷ 2 = 3)	3	
2.1.6	toString: Stuur String terug (1), met al die velde (2), roep berekenPunte metode (1), skakel om na integer waar nodig(1), tabs (1) (6 ÷ 2 = 3)	3	
2.2			
2.2.1	Uses klas (1) Verklaar skikking van objekte(1) Verklaar teller(1), Verklaar tekslêer(1), if file not exists(1) then showMessage(1), exit(1) Voor lus (while loop): Assignfile (1), Reset(1) Inisialiseer teller(1), while (1) Binne while: Inc teller (1) Lees uit lêer (1), breek string op met pos van #(1), kopieer dag uit(1), delete(1), haal etenstyd uit(2), haal vette uit(1), skakel om na int(1), haal proteïene uit (1) en koolh(1), skep objek (1) voeg objek by in regte posisie in skikking(1) met argumente(1) Maak lêer toe buite lus(1) (26 ÷ 2 = 13)	13	
2.2.2	Tik dag in (1), initialiseer totaal(1), Vertoon opskrifte(2), for lus(1) toets item uit skikking deur getDag metode te roep(1) en dit te vergelyk met gebruiker se toevoer(1) Binne if: Vertoon met toString(1), roep berekenPunte(1) en tel by totaal (1), buite for, vertoon totaal(1) as totaal > 50(1) vertoon boodskap(1) anders vertoon ander boodskap(1) (14 ÷ 2 = 7)	7	
2.2.3	Vertoon opskrifte(2)Lus(1) toets skikkingselement(1) en roep geenVette metode(1) en vertoon skikkingselement binne if (1) (6 ÷ 2 = 3)	3	
2.2.4	Opskrif(1), Inisialiseer twee veranderlikes vir maks en min punte(2) lus(1), roep berekenPunte metode (1) en toets of minder as minste(1), binne die if (1) dateer minste op(1) dateer minste dag op (1), Herhaal die if-stelling vir Makspunte en maksDag met > (1) Buite lus: vertoon inligting van Hoogste(1) en Laagste(1) (12 ÷ 2 = 6)	6	
	TOTAAL:	44	

BYLAAG D – Maart 2010**VRAAG 3: DELPHI PROGRAMMERING**

SENTRUMNOMMER:.....		EKSAMENNOMMER:	
VRAAG 3 DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
3.1	Subprogram: 6 punte (1) Definieer subprogram (1) Initialiseer nuwe veranderlike (1) Stap met lus deur selnommer (1) As nie spasie dan (1) Voeg karakter by nuwe veranderlike (1) Stuur veranderlike terug Haal moontlike wenners uit: 13 punte (1) Stap met lus deur gegewe skikking (2) Kry selnommer uit string (1) Roep funksie om spasie te verwyder (2) Kry antwoord uit string (1) As die antwoord reg is (1) ignoreer hoofletters Verhoed duplikate :3 punte (1) Verklaar nuwe skikking (1) gebruik teller vir nuwe skikking reg (1) voeg nommer in nuwe skikking in OF gebruik gegewe skikking deur te toets vir duplikate in die huidige skikking Vertoon inligting: opskrif, lus(1), vertoon(1)	19	
3.2	Kies en Vertoon Wenners: 10 punte (2) Genereer random getal (1) herhaal 3 keer / gebruik 3 veranderlikes Enige metode om wenners wat reeds gekies is nie weer te kies nie: 4 punte. Een moontlike oplossing: Skep nuwe skikking(1), Initialiseer en gebruik nuwe teller(1), Toets(1), plaas in nuwe skikking indien nog nie gekies nie(1) Vertoon opskrif(1), wenners (1) in korrekte formaat(1)	10	
3.3	Stoor Wenners: 7 punte (1) Gebruik regte naam vir lêer (1) Maak lêer oop om daarheen te skryf (1) Skryf die opskrif na die lêer (1) Skryf 3 items na tekslêer(1) items moet in regte formaat wees(1) (1) Maak lêer toe	7	
	TOTAAL:	36	

BYLAAG E – Maart 2010**VRAAG 1: JAVA – PROGRAMMERING EN DATABASIS**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG 1: JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS PUNTE	LEERDER SE PUNTE
1.1	SELECT✓ *✓ FROM tblMense✓ ORDER BY✓ InskryfDatum DESC✓	5	
1.2	SELECT PersoonID, Naam, B_Gewig, D_Gewig, ✓ ROUND✓(B_Gewig – D_Gewig✓, 1) ✓ AS KgsOmTeVerloor✓ FROM tblMense✓	6	
1.3	DELETE✓ FROM tblMense✓ WHERE✓ Balans > 400✓ AND✓ Naam <>'Uriel Knight'✓	6	
1.4	SELECT MIN(Gewig) ✓ AS MinGewig✓, PersoonID✓ FROM tblWeegin✓ GROUP BY✓ PersoonID✓	6	
1.5	Sleutelbord-toevoer van gebruiker ✓ SELECT tblMense.PersoonID✓, Naam, D_Gewig, WeegDatum✓, Gewig FROM tblMense✓, tblWeegin✓ WHERE tblMense.PersoonID✓ = tblWeegin.PersoonID✓ AND✓ tblMense.PersoonID✓ = " + id✓	10	
1.6	UPDATE tblWeegin✓ SET✓ Gewig✓ = Gewig✓ * 1.1✓ WHERE MONTH(WeegDatum) ✓ = 5✓	7	
	TOTAAL:	40	

BYLAAG F – Maart 2010

VRAAG 2: JAVA – OBJEK-GEORIENTEERDE PROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG 2 JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
2.1			
2.1.1	Definieer klas (1) EteXXXX , vyf privaat (1) velde , 2 strings(1), 3 integers(1), Metodes public (1) Konstruktor (1) (6 ÷ 2 = 3)	3	
2.1.2	Konstruktor: Parameters in regte volgorde (1), regte tipes (1) toegeken aan velde (2) (4 ÷ 2 = 2)	2	
2.1.3	getDag (1) metode, regte tipe teruggestuur (1) regte veld (1) teruggestuur (1) (4 ÷ 2 = 2)	2	
2.1.4	geenVette metode: stuur Boolean terug(1) Toets of vette 0 is(1) stuur true terug(1) anders stuur false terug(1) (4 ÷ 2 = 2)	2	
2.1.5	berekenPunte: stuur heelgetal terug (1), regte berekeninge (2) tel 10 punte by vir meer as 2 vette (1) trek 2 punte af vir geen vette (1) stuur total terug (1) (6 ÷ 2 = 3)	3	
2.1.6	toString: Stuur String terug (1), met al die velde (2), roep berekenPunte metode (1), skakel om na integer waar nodig(1), tabs (1) (6 ÷ 2 = 3)	3	
2.2			
2.2.1	Verklaar skikking van objekte(2) Verklaar en inisialiseer teller(2), Verklaar tekslêer objek(1) met regte naam vir lêer(1) if file not exists(1) dan skep Scanner / FileReader objek(1) While lus(1) Binne while: Lees uit lêer (1), breek string op met pos van #(1) in token skikking(2), kry dag(1), kry etenstyd uit(1), kry vette(1), skakel om na int(1), kry proteiene uit (1) en koolh(1), skep objek (1) voeg objek by in regte posisie in skikking(1) met argumente(1) Inc teller (1) Maak lêer toe buite lus(1) buite lus As lêer nie bestaan nie, vertoon boodskap(1) en stop(1) (26 ÷ 2 = 13)	13	
2.2.2	Tik dag in (1), initialiseer totaal(1), Vertoon opskrifte(2), for lus(1) toets item uit skikking(1) deur getDag metode te roep(1) en dit te vergelyk met gebruiker se toevoer(1) Binne if: Vertoon met toString(1), roep berekenPunte(1) en tel by totaal (1), na die for vertoon totaal(1) as totaal > 50(1) vertoon boodskap(1) anders vertoon ander boodskap(1) (14 ÷ 2 = 7)	7	
2.2.3	Vertoon opskrifte(2) Lus(1) toets skikkingselement(1) en roep geenVette metode(1) en vertoon skikkingselement binne if (1) (6 ÷ 2 = 3)	3	
2.2.4	Opskrif(1), Initialiseer twee veranderlikes vir maks en min punte(1) en maks en min dag(1) lus(1), roep berekenPunte metode (1) toets of minPunte minder is as minste(1) Binne if(1) dateer minste(1) en dag (1) op. Herhaal if-stelling met maksPunte en maksdag met >(1) Buite lus: vertoon inligting van Hoogste(1) en Laagste(1) (12 ÷ 2 = 6)	6	
	TOTAAL:	44	

BYLAAG G – Maart 2010**VRAAG 3: JAVA PROGRAMMERING**

SENTRUMNOMMER:.....		EKSAMENNOMMER:	
VRAAG 3 JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
3.1	Subprogram: 6 punte (1) Definieer subprogram (1) Initialiseer nuwe veranderlike (1) Stap met lus deur selnommer (1) As nie spasie dan (1) voeg karakter by nuwe veranderlike (1) stuur veranderlike terug Haal moontlike wenners uit: 13 punte (1) Stap met lus deur gegewe skikking (2) Kry selnommer uit string (1) Roep funksie om spasie te verwyder (2) Kry antwoord uit string (1) As die antwoord reg is (1) ignoreer hoofletters Verhoed duplikate: 3 punte (1) Verklaar nuwe skikking (1) gebruik teller vir nuwe skikking reg (1) voeg nommer in nuwe skikking in OF gebruik gegewe skikking deur te toets vir duplikate in die huidige skikking Vertoon inligting: opskrif, lus(1), vertoon(1)	19	
3.2	Kies en Vertoon Wenners: 10 punte (2) Genereer random getal (1) herhaal 3 keer / gebruik 3 veranderlikes Enige metode om wenners wat reeds gekies is nie weer te kies nie: 4 punte. Een moontlike oplossing: Skep nuwe skikking(1), Initialiseer en gebruik nuwe teller(1), Toets(1), plaas in nuwe skikking indien nog nie gekies nie(1) Vertoon opskrif(1), wenners (1) in korrekte formaat(1)	10	
3.3	Stoor Wenners: 7 punte (1) Gebruik regte naam vir lêer (1) Maak lêer oop om daarheen te skryf (1) Skryf die opskrif na die lêer (1) Skryf 3 items na tekslêer(1) items moet in regte formaat wees(1) (1) Maak lêer toe	7	
	TOTAAL:	36	