



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

NASIONALE SENIOR SERTIFIKAAT

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

NOVEMBER 2010

MEMORANDUM

PUNTE: 120

Hierdie memorandum bestaan uit 26 bladsye.

Algemene inligting:

- **Bladsye 2 – 11 bevat die Delphi memoranda met moontlike oplossings vir VRAE 1 tot 3 in programmeringskode.**
- **Bladsye 12 – 20 bevat die Java memoranda met moontlike oplossings vir VRAE 1 tot 3 in programmeringskode.**
- **Bladsye 21 – 26 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****unit Vraag1_UXXXX;**

```
interface
```

```
uses
```

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

```
type
```

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

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
frmWaarnemings: TfrmWaarnemings;
```

```
implementation
```

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

```

procedure TfrmWaarnemings.btnAClick(Sender: TObject);           // Vraag 1.1
begin
    qryWaarnemings.Active := False;
    qryWaarnemings.SQL.Text := 'SELECT * ✓FROM tblWaarnemings✓ ORDER BY✓
    WaarneemID DESC✓';
    qryWaarnemings.ExecSQL;
    qryWaarnemings.Open;
end;                                                             (4)
//=====

procedure TfrmWaarnemings.btnBClick(Sender: TObject);           // Vraag 1.2
begin
    qryWaarnemings.Active := False;
    qryWaarnemings.SQL.Text := 'SELECT DISTINCT✓ Dier✓ FROM tblWaarnemings✓
    WHERE Jonk = true✓';
    qryWaarnemings.ExecSQL;
    qryWaarnemings.Open;
end;                                                             (4)
//=====

procedure TfrmWaarnemings.btnCClick(Sender: TObject);           // Vraag 1.3
begin
    qryWaarnemings.Active := False;
    qryWaarnemings.SQL.Text := 'SELECT VeldwagterID, Naam, Van, ✓ year(Now())✓
    - ✓year (DatumAangestel) ✓ AS [TotaleJare] ✓ FROM tblVeldwagters'✓';
    qryWaarnemings.ExecSQL;
    qryWaarnemings.Open;
end;                                                             (6)
//=====

procedure TfrmWaarnemings.btnDClick(Sender: TObject);           // Vraag 1.4
begin
    qryWaarnemings.Active := False;
    qryWaarnemings.SQL.Text := 'SELECT Dier✓, format(Avg(AantalDiere) ✓,"0.00"✓)
    AS [GemWaarnemings] ✓ FROM tblWaarnemings✓ GROUP BY Dier'✓';
    qryWaarnemings.ExecSQL;
    qryWaarnemings.Open;
end;                                                             (6)
//=====

procedure TfrmWaarnemings.btnEClick(Sender: TObject);           // Vraag 1.5
var
    id : integer;
    aantRekords : integer;
begin
    id := StrToInt(InputBox('Uitvee van waarneming', 'Tik die ID in van die
    waarneming wat uitgevee moet word', '1')); ✓
    qryWaarnemings.Active := False;
    qryWaarnemings.SQL.Text := 'DELETE✓ FROM tblWaarnemings✓ WHERE WaarneemID =
    '+ IntToStr(id); ✓
    aantRekords := qryWaarnemings.ExecSQL;
    MessageDlg (IntToStr(aantRekords) + ' rekord is
    uitgevee.',mtInformation,[mbok],0);
end;                                                             (4)
//=====

```

```
procedure TfrmWaarnemings.btnFClick(Sender: TObject);           // Vraag 1.6
var
  aantRekords : integer;
begin
  gryWaarnemings.Active := False;
  gryWaarnemings.SQL.Text := 'UPDATE✓ tblWaarnemings✓ SET✓ Dier =
    "Witrenoster" ✓WHERE Dier = "Renoster"✓';
  aantRekords := gryWaarnemings.ExecSQL;
  MessageDlg (IntToStr(aantRekords) + ' rekords is opgedateer.' ,
    mtInformation ,[mbok],0);
end;                                                            (5)
//=====

procedure TfrmWaarnemings.btnGClick(Sender: TObject);           // Vraag 1.7
begin
  gryWaarnemings.Active := False;
  gryWaarnemings.SQL.Text := 'SELECT WaarneemDatum,Naam,Van✓ FROM
tblWaarnemings, tblVeldwagters✓ WHERE tblWaarnemings.VeldwagterID✓ =
tblVeldwagters.VeldwagterID ✓AND Dier = "Olifant" ✓AND WaarneemDatum >
#30/04/2010#'; ✓
  gryWaarnemings.ExecSQL;
  gryWaarnemings.Open;
end;                                                            (6)
                                                            [35]

end.
```

VRAAG 2: OBJEK-GEÖRIËNTEERDE PROGRAMMERING**uDeelnemerXXXX;**

interface

uses SysUtils;

// V 2.1.1

type TDeelnemer = class

```

private✓
    naam : String; ✓
    grootWildTel : integer;
    kleinWildTel : integer; ✓
    geVeerdesTel : integer;

```

(3)

public

```

    constructor Create; overload;
    constructor Create(sNaam : String); overload;
    function toString : String;
    function aantalDiere : integer;
    procedure sienGroot;
    procedure sienKlein;
    procedure sienGeveerde;
    function berekenPunte : integer;
    function getNaam : String;
    function meesteGesien: String;

```

end;

implementation

{ Deelnemer }

constructor TDeelnemer.Create;

begin

end;

//=====

constructor TDeelnemer.Create;

begin

end;

// V 2.1.2

(3)

constructor TDeelnemer.Create(sNaam : String); ✓

begin

```

    naam := sNaam;
    grootWildTel := 0;
    kleinWildTel := 0;
    geVeerdesTel := 0;

```

✓

end;

procedure TDeelnemer.sienGroot;

begin

inc(grootWildTel);

end;

procedure TDeelnemer.sienKlein;

begin

inc(kleinWildTel);

end;

procedure TDeelnemer.sienGeveerde;

begin

inc(geVeerdesTel);

Kopiereg voorbehou

V 2.1.1

(1) Vier privaat veranderlikes

(1) Verklaar naam as private String

(1) Al die tel veranderlikes verklaar as privaat heelgetalle

V 2.1.2

(a) (1) Ontvang naam as String

(1) Initialiseer alle 'instance fields'

(b) (1) Verwyder kommentaar-teken uit kode in die drie gegewe 'sien'-metodes

```
end;
//=====
```

```
// V 2.1.3 (3)
function TDeelnemer.berekenPunte: integer;
begin
    Result := grootWildTel * 5 + kleinWildTel * 3 + geVeerdesTel * 2; ✓✓✓;
```

V 2.1.3

(3) Vermenigvuldig elke teller met die regte waardes, tel bymekaar & ken toe aan Result. Trek 1 punt af vir elke tipe fout, nie vir dieselfde tipe fout nie.

```
end;
//=====
// V 2.1.4 (2)
function TDeelnemer.aantalDiere: integer✓;
begin
    Result := grootWildTel + kleinWildTel + geVeerdesTel; ✓
end;
```

V 2.1.4

(1) Stuur tipe integer terug
(1) Stuur die som van die diere-teller terug

```
// V 2.1.5 (2)

function TDeelnemer.getNaam: String; ✓
begin
    Result := naam; ✓
end;
```

V 2.1.5

(1) Stuur tipe String terug
(1) Stuur naam terug

```
// V 2.1.6 (4)
function TDeelnemer.meesteGesien: String✓;
begin
    if (grootWildTel > kleinWildTel) AND (grootWildTel > geVeerdesTel) ✓ then
        Result := 'Grootwild'
    else if (grootWildTel < kleinWildTel) AND (kleinWildTel > geVeerdesTel) ✓
then
        // Klein wild EN voëls moet else-stelling hê, -
        Result := 'Kleinwild' // anders nog 'n if-stelling vir voëls
    else✓
        Result := 'Voëls';
end;

    // Korrekte variasies van hierdie kode moet aanvaar word
//=====
```

V 2.1.6

(1) Stuur tipe String terug
(1) If om Groot wild te kry
(1) If om Klein wild te kry
(1) kry Geveerdes

```
// V 2.1.7 (5)
function TDeelnemer.toString: String;
var
    ObjStr : String;
begin
    ObjStr := 'Deelnemer : ' + naam + #13✓;
    ObjStr := ObjStr ✓+ 'Groot : ' + IntToStr(grootWildTel)
        + ' Klein : ' + IntToStr(kleinWildTel) + ' Voëls : ' +
        IntToStr(geVeerdesTel) + #13✓;
    ObjStr:= ObjStr+ 'Totale aantal diere : ' + #9✓+ IntToStr(aantalDiere) ✓;
    Result := ObjStr;
end;
// Kan meer tabs (#9) gebruik as wat hier aangedui is
//=====
```

V 2.1.7

(1) Nuwe reël (#13)
(1) Voeg stringe saam
(1) Alle tel elemente bygevoeg
(1) Voeg laaste reël by
(1) Tab (#9)

unit VraagTweeXXXX_U;

```
interface
```

```
uses
```

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

```
Kopiereg voorbehou
```

```
Blaai om asseblief
```

```

type
  TfrmVraag2 = class(TForm)
    redAfvoer: TRichEdit;

    MainMenu1: TMainMenu;
    mnuAVraag2: TMenuItem;
    mnuBVraag2: TMenuItem;
    Quit1: TMenuItem;
    procedure Quit1Click(Sender: TObject);
    procedure mnuAVraag2Click(Sender: TObject);
    procedure mnuBVraag2Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag2: TfrmVraag2;

implementation

{$R *.dfm}
// V 2.2.1
uses
  uDeelnemerXXXX; ✓
var
  Deelnemer : TDeelnemer; ✓

procedure TfrmVraag2.Quit1Click(Sender: TObject);
begin
  Application.Terminate;
end;

procedure TfrmVraag2.FormActivate(Sender: TObject);
var
  tLeer : textfile;
  sNaam, sDier, sDiereNaam : String;
  iGeldig, iOngeldig, iHakie : integer;
  sLetter : char;
begin
  redAfvoer.Lines.Clear;
  iGeldig:= 0;
  iOngeldig := 0; } ✓

  If FileExists('Waarnemings.txt') <> TRUE then ✓
  begin
    ShowMessage('Lêer nie gevind'); } ✓
    Application.Terminate;
  end
  AssignFile(tLeer, 'Waarnemings.txt'); ✓
  Reset(tLeer); ✓
  ReadLn(tLeer, sNaam); ✓
  Deelnemer := TDeelnemer.Create ✓ (sNaam); ✓
  while NOT EOF(tLeer) DO ✓
  begin
    ReadLn(tLeer, sDier); ✓
    iHakie := pos('(', sDier);
    sDiereNaam := copy(sDier,1,iHakie - 1); ✓
    sLetter := sDier[iHakie + 1]; ✓
  end;
end;

```

(20)**V 2.2.1**

- (1) uses object klas
- (1) Verklaar objek veranderlike
- (1) Inisialiseer twee tel veranderlikes
- (1) Toets of lêer bestaan
- (1) Indien nie, Vertoon boodskap & termineer
- (1) 'Assign' file
- (1) Maak lêer oop vir lees
- (1) Lees naam van deelnemer uit lêer
- (2) Skep deelnemer objek
- (1) Lus met krulhakies in regte plekke
- (1) Lees reël uit teksleër
- (1) Kopieer dier se naam uit reël
- (1) Kopieer dier kategorie uit reël
- (1) Case OF if-stellings
 - Vir elke geldige kategorie
 - (1) Roep metode om teller te vermeerder
 - (1) Tel 1 by teller van geldige kategorieë
- (1) Vir ongeldige kategorie
 - (1) inc teller & vertoon boodskap
- (1) Vertoon die aantal geldige &

```

        case sLetter of✓
        {
            'G' : begin
                Deelnemer.sienGroot;
                inc(iGeldig);
                end;
            'K' : begin
                Deelnemer.sienKlein;
                inc(iGeldig);
                end;
            'V' : begin
                Deelnemer.sienGeveerde;
                inc(iGeldig);
                end;
        } ✓✓✓
        else begin
            redAfvoer.Lines.Add(sDiereNaam + ' is nie in ' + n
                                geldige kategorie nie'); ✓
            Inc(iOngeldig);
            end;
        end; // case
    end; // while
    CloseFile(tLeer);
    redAfvoer.Lines.Add('');
    redAfvoer.Lines.Add(IntToStr(iGeldig) + ' geldige kategorieë is
                                verwerk');
    redAfvoer.Lines.Add(IntToStr(iOngeldig) + ' ongeldige kategorieë is
                                verwerk'); } ✓

end;
//=====
// V 2.2.2 (2)
procedure TfrmVraag2.mnuAVraag2Click(Sender: TObject);
begin
    redAfvoer.Lines.Clear;
    redAfvoer.Lines.Add(Deelnemer.toString); ✓✓
end;
//=====
// V 2.2.3 (5)
procedure TfrmVraag2.mnuBVraag2Click(Sender: TObject)
var
    tLeer : textfile;
    leerNaam : String;
begin
    leerNaam := Deelnemer.getNaam + '.txt'; ✓

    AssignFile(tLeer, leerNaam); } ✓
    Rewrite(tLeer);

    Writeln(tLeer, 'Deelnemer : ' + Deelnemer.getNaam);
    Writeln(tLeer, 'Totale aantal diere : ' + IntToStr(Deelnemer.aantalDiere)); ✓✓
    Writeln(tLeer, 'Punte : ' + IntToStr(Deelnemer.berekenPunte));
    Writeln(tLeer, 'Kategorie wat die meeste waargeneem is : ' +
                                Deelnemer.meesteGesien); } ✓

    CloseFile(tLeer); ✓

    redAfvoer.Lines.Clear;
    redAfvoer.Lines.Add('Resultate Suksesvol na Lêer Geskryf');
end;

end.
//=====

```

V 2.2.2

- (1) Roep die toString-metode van die objek
- (1) in 'n vertoon-stelling

V 2.2.3

- (1) Stel die naam van die nuwe lêer reg saam
- (1) Maak die nuwe lêer oop
- (1) Stel die inligting reg saam om na die lêer te skryf
- (1) Skryf na die lêer
- (1) Maak lêer toe

VRAAG 3: DELPHI PROGRAMMERING

NB: Hierdie is slegs 'n voorbeeld – leerder kan hierdie vraag op enige manier beantwoord soos hulle wil. Maak gebruik van die algemene rubriek wat ingesluit is by die nasienblaaie vir nasien.

```

unit Vraag3XXXX_U;

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);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag3: TfrmVraag3;

implementation

{$R *.dfm}
//Vraag 3.1 (3)

var
  arrData : array[1..12] ✓ of string; ✓

procedure TfrmVraag3.FormCreate(Sender: TObject);
begin
  arrData[1] := '12,15:02h00';
  arrData[2] := '13,10:05h00';
  arrData[3] := '9,20:06h00';
  arrData[4] := '10,15:09h00';
  arrData[5] := '7,8:10h00';
  arrData[6] := '10,10:11h00';
  arrData[7] := '12,18:14h00';
  arrData[8] := '7,18:17h00';
  arrData[9] := '11,7:19h00';
  arrData[10] := '10,10:20h00';
  arrData[11] := '2,1:23h00';
  arrData[12] := '12,17:24h00'; ✓
end;
//=====

```

V 3.1

(2) Verklaar skikking van stringe
(1) Verwyder kommentaartekens van
gegewe stellings wat stringe aan die
skikking toeken

//Vraag 3.2 (17)

```
procedure TfrmVraag3.btnAClick(Sender: TObject);
var
```

```
    K, iAfstand :integer;
    iWaterX, iWaterY, xPos, yPos :integer;
    iKomma, iDPunt :integer;
    sTyd :string;
```

```
begin
```

```
    redAfvoer.Paragraph.TabCount := 4;
    redAfvoer.Paragraph.Tab[0] := 60;
    redAfvoer.Paragraph.Tab[1] := 100;
    redAfvoer.Paragraph.Tab[2] := 130;
    redAfvoer.Paragraph.Tab[2] := 150;
    iWaterX := 10; }
    iWaterY := 10; }
```

V 3.2

(1) Initialiseer water se x & y posisies

(2) Vertoon opskrif & onderopskrifte

(1) Lus met begin & end op regte

plekke

Binne die lus:

(2) Onttrek x-pos,

(2) Onttrek y-pos,

(2) Onttrek tyd uit string van skikking

(3) Bereken afstand,

(1) Rond antwoord af,

(3) Vertoon alle inligting

```
    redAfvoer.Lines.Add('Afstande van die watergat af'); ✓
```

```
    redAfvoer.Lines.Add('Tyd' + #9 + 'Afstand(km)' + #9 + 'X-pos' + #9 + 'Y-pos'); ✓
```

```
    for K := 1 to 12 do ✓
```

```
        begin
```

```
            iKomma := pos(',', arrData[K]); ✓
```

```
            xPos := StrToInt(copy(arrData[K],1,iKomma-1)); ✓
```

```
            iDPunt := pos(':', arrData[K]); ✓
```

```
            yPos := StrToInt(copy(arrData[K],iKomma + 1,iDPunt - iKomma-1)); ✓
```

```
            iAfstand := Round( Sqrt( Sqr(xPos - iWaterX ) + Sqr(yPos - iWaterY ) )); ✓
```

```
            delete(arrData[K],1,iDPunt); ✓
```

```
            sTyd := arrData[K]; ✓
```

```
            redAfvoer.Lines.Add(sTyd + #9 + ' ' + FloatToStr(iAfstand) + #9 + #9 + #9 + IntToStr(xPos) + #9 + ' ' + IntToStr(yPos)); ✓
```

```
        end;
    end;
```

```
end;
```

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

//Vraag 3.3 (16)

```
procedure TfrmVraag3.btnBClick(Sender: TObject);
```

```
var
```

```
    K, iAantal, iTel, iLengte :integer;
    iRandomGetal, L :integer;
    sDier, sEtiket :string;
```

```
begin
```

```
    redAfvoer.Paragraph.TabCount := 1;
```

```
    redAfvoer.Paragraph.Tab[0] := 90;
```

```
    Randomize;
```

```
    redAfvoer.Clear;
```

```
        iAantal := StrToInt(InputBox('Diere Etiket',
                                     'Tik die aantal verskillende diersoorte
                                     in die groep in', '')); ✓
```

```
    for K := 1 to iAantal do ✓
```

```
        begin ✓
```

```
            sDier := InputBox('Diere Etiket',
                              'Tik die naam van diersoort ' +
                              IntToStr(K) + ' in', ''); ✓
```

```
            iTel := StrToInt(InputBox('Diere Etiket',
                                      'Tik die aantal van diersoort ' +
                                      IntToStr(K) + ' in die groep in', '')); ✓
```

V 3.3

(1) Tik aantal verskillende tipes diere in,

(1) Lus van 1 tot die aantal verskillende tipes diere,

(1) Binne lus

(1) Tik naam van diersoort in

(1) Tik die aantal in die groep in van diersoort binne lus,

(1) Onttrek eerste 2 letters en ken toe aan etiket

(2) Onttrek die laaste letter,

(2) Genereer random getal in regte reeks,

(1) toets vir ewegetal,

(1) Voeg random getal by etiket,

(1) Opskrif

(1) Binneste lus om nommers van diere binne die groep by etiket te voeg

(2) Vertoon die korrekte data reg gespasieer binne die binneste lus

```

sEtiket := copy(sDier, 1, 2); ✓
iLengte := length(sDier); ✓
sEtiket := sEtiket + copy(sDier, iLengte,1); ✓
repeat
    iRandomGetal := Random(900) ✓ + 100; ✓
until iRandomGetal mod 2 = 0; ✓
redAfvoer.Lines.Add(' ');

sEtiket := sEtiket + IntToStr(iRandomGetal ); ✓
redafvoer.Lines.Add(sDier + #9 + 'Etiketnommer'); ✓
for L := 1 to iTel do ✓
    begin
        redAfvoer.Lines.Add(intToStr(L) + '.' + #9 + sEtiket ✓ + '-' +
                                intToStr(L) ✓);
    end;
end;
end;
//=====
end.

```

OF
iRandomNumber:= Random(899)+ 100;
if iRandomNumber mod 2 <> 0 then
 inc (iRandomNumber);

[36]**EINDE VAN AFDELING A: DELPHI****TOTAAL AFDELING A: 120**

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

```

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

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

        Waarneming DB = new Waarneming();
        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 tblWaarnemings✓ ORDER BY✓ WaarneemID DESC✓";
                        DB.A(sql);
                        break;
                    }
                    (4)
                //=====
                case 'B':
                    // Vraag 1.2
                    {
                        sql = "SELECT DISTINCT✓ Dier✓ FROM tblWaarnemings✓
                            WHERE Jonk = true";✓
                        DB.B(sql);
                        break;
                    }
                    (4)
                //=====
                case 'C':
                    // Vraag 1.3
                    {
                        sql = "SELECT VeldwagterID, Naam, Van, ✓ year(Now())✓ - ✓year
                            (DatumAangestel) AS [TotaleJare] ✓ FROM tblVeldwagters";✓
                        DB.C(sql);
                        break;
                    }
                    (6)
            }
        }
    }
}

```

```

        case 'D':
            // Vraag 1.4
            {
                sql = "SELECT Dier✓, format(Avg(AantalDiere) ✓,'0.00'✓)AS
                    [GemWaarnemings] ✓FROM tblWaarnemings ✓GROUP BY Dier✓";

                DB.D(sql);
                

OF round(Avg(AantalDiere),2)


                break;
            }
            //=====
            case 'E':
                // Vraag 1.5
                {
                    Scanner kb = new Scanner (System.in);
                    System.out.println();
                    System.out.println("Tik die ID in van die waarneming wat uitgegee moet
                        word ");

                    int id = kb.nextInt();✓

                    sql = "DELETE✓ FROM tblWaarnemings✓ WHERE WaarneemID = "+ id +"✓";
                    DB.E(sql);
                    break;
                }
                //=====
            case 'F':
                // Vraag 1.6
                {
                    sql = "UPDATE✓ tblWaarnemings✓ SET✓ Dier = 'Witrenoster'✓
                        WHERE Dier = 'Renoster'";✓

                    DB.F(sql);
                    break;
                }
                //=====
            case 'G':
                // Vraag 1.7
                {
                    sql = "SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings,
                        tblVeldwagters ✓WHERE tblWaarnemings.VeldwagterID = ✓
                        tblVeldwagters.VeldwagterID✓ AND Dier = 'Olifant'✓ AND
                        WaarneemDatum > #30/04/2010#"; ✓

                    DB.G(sql);
                    break;
                }
                //=====

        } // switch
    }while (keuse != 'V');

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

```

[35]

VRAAG 2: OBJEK-GEÖRIËNTEERDE PROGRAMMERING**DeelnemerXXXX.java**

```
public class DeelnemerXXXX
{
```

```
    // V 2.1.1. (3)
    private ✓ String naam; ✓
    private int grootWildTel;
    private int kleinWildTel;
    private int geVeerdesTel; } ✓
```

```
    public DeelnemerXXXX ()
    {
    }
}
```

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

```
    // V 2.1.2 (3)
    public DeelnemerXXXX (String aNaam) ✓
    {
        naam = aNaam;
        grootWildTel = 0;
        kleinWildTel = 0;
        geVeerdesTel = 0; } ✓
    }
```

```
    public void sienGeveerde()
    {
        geVeerdesTel++;
    }

    public void sienKlein()
    {
        kleinWildTel++;
    }

    public void sienGroot()
    {
        grootWildTel ++;
    }
```

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

```
    // V 2.1.3 (3)
```

```
    public int berekenPunte()
    {
        return grootWildTel * 5 + kleinWildTel * 3 + geVeerdesTel * 2 ✓✓✓;
    }
}

//=====
```

V 2.1.1

- (1) Vier privaat veranderlike
- (1) Verklaar naam as private String
- (1) Al die tel veranderlikes verklaar as privaat heelgetalle

V 2.1.2

- (1) Ontvang naam as String
- (1) Initialiseer alle 'instance fields'
- (1) Verwyder kommentaar-teken uit kode in die drie gegewe sien-metodes

V 2.1.3

- (3) Vermenigvuldig elke kategorie met die regte waarde, tel bymekaar en stuur waarde terug
- Tel 1 punt af vir elke een tipe fout, nie vir dieselfde tipe fout nie

// V 2.1.4

(2)

```
public int✓ aantalDiere()
{
    return grootWildTel + kleinWildTel + geVeerdesTel; ✓
}
```

V 2.1.4

- (1) Stuur tipe int terug
- (1) Stuur die som van die diere-teller terug

//=====

// V 2.1.5

(2)

```
public String getNaam()✓
{
    return naam; ✓
}
```

V 2.1.5

- (1) Stuur tipe String terug
- (1) Stuur naam terug

//=====

// V 2.1.6

(4)

```
public String✓ meesteGesien()
{
    if (grootWildTel > kleinWildTel & grootWildTel > geVeerdesTel) ✓
    {
        return "Groot Wild";
    }
    else if (kleinWildTel > grootWildTel & kleinWildTel > geVeerdesTel)✓
    {
        return "Klein Wild";
    }
    else✓
    {
        return "Voël";
    }
    // Klein wild EN voëls moet else-stelling hê, -
    // anders nog 'n if-stelling vir voëls
    // Korrekte variasies van hierdie kode moet aanvaar word
}
```

V 2.1.6

- (1) Stuur tipe String terug
- (1) If om Groot wild te kry
- (1) If om Klein wild te kry
- (1) Kry geveerdes

//=====

// V 2.1.7

(5)

```
public String toString()
{
    String objStr = "Naam : " + naam + "\n"✓;
    objStr = objStr ✓+ "Groot : " + grootWildTel + " Klein : "
        + kleinWildTel + " Voëls : " + geVeerdesTel + "\n"✓;
    objStr = objStr + "Totale aantal diere: \t"✓ + aantalDiere();✓
    return objStr;
}
```

V 2.1.7

- (1) Nuwe reël (\n)
- (1) Voeg stringe saam
- (1) Alle tel elemente bygevoeg
- (1) Voeg laaste reël by
- (1) Tab (\t) added

```
} // aanvaar korrekte gebruik van 'formatter' om die reël saam te stel
```

//=====

ToetsDeelnemerXXXX.java

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

```
public class ToetsDeelnemerXXXX
{
```

// V 2.2.1**(20)**

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

```
{
    DeelnemerXXXX Deelnemer = new DeelnemerXXXX (); ✓

    File f = new File("Waarnemings.txt"); ✓
    if (!f.exists()) ✓
    {
        System.out.println("Lêer nie gevind nie"); } ✓
        System.exit(0);
    }

    Scanner sc = new Scanner(f); ✓

    String sNaam = sc.nextLine(); ✓
    Deelnemer ✓ = new DeelnemerXXXX ✓ (sNaam ✓);

    int geldig = 0; } ✓
    int ongeldig = 0; }

    while (sc.hasNextLine()) ✓
    {
        String sDier = sc.nextLine(); ✓
        int hakie = sDier.indexOf('(');
        String dier = sDier.substring(0, hakie); ✓
        char letter = sDier.charAt(hakie + 1); ✓

        switch (letter) ✓
        {
            case 'G' : Deelnemer.sienGroot();
                       geldig++;
                       break;
            case 'K' : Deelnemer.sienKlein();
                       geldig++;
                       break;
            case 'V' : Deelnemer.sienGeveerde();
                       geldig++;
                       break;
            default: System.out.println(dier + " is nie in 'n geldige kategorie
                                                                nie");
                       ongeldig++; ✓
                       break;
        }

        System.out.println();
        System.out.println(geldig + " geldige kategorieë verwerk"); } ✓
        System.out.println(ongeldig + " ongeldige kategorieë verwerk"); }

    sc.close();
}
```

V 2.2.1

- (2) Verklaar objek veranderlike
- (1) Inisialiseer twee tel veranderlikes
- (1) Toets of lêer bestaan
- (1) Vertoon boodskap & termineer
- (1) 'Assign' file
- (1) Maak lêer oop vir lees
- (1) Lees naam van deelnemer uit lêer
- (3) Skep deelnemer objek
- (1) Lus met krulhakies in regte plekke
- (1) Lees reël uit teksleër
- (1) Kopieer dier se naam uit reël
- (1) Kopieer dier kategorie uit reël
- (1) switch OF if-stellings
- (1) Vir elke geldige kategorie
 - (1) roep metode om teller te vermeerder
 - (2) tel 1 by teller van geldige kategorieë
- Vir ongeldige kategorie
 - (1) inc teller & vertoon boodskap
- (1) Vertoon die aantal geldige & ongeldige kategorieë


```
//=====
BufferedReader inKb = new BufferedReader (new InputStreamReader (System.in));
char ch = ' ';
while (ch != 'V')
{
    System.out.println();
    System.out.println("          KEUSELYS");
    System.out.println(" ");
    System.out.println("          Opsie A");
    System.out.println("          Opsie B");
    System.out.println(" ");
    System.out.println("          V - VERLAAT");
    System.out.println(" ");
    System.out.print("          Jou keuse? ");
    ch = inKb.readLine().toUpperCase().charAt(0);

    switch (ch)
    {
// V 2.2.2                                (2)
    case 'A':
    {
        System.out.println();
        System.out.println(Deelnemer.toString());✓✓
        break;
    }
}
//=====
// V 2.2.3                                (5)
    case 'B':
    {
        PrintWriter fUit = new PrintWriter(new File✓ (Deelnemer.getNaam()+
                                                    ".txt"));✓

        fUit.println("Deelnemer : " + Deelnemer.getNaam());
        fUit.println("Totale aantal diere : " + Deelnemer.totaleDiere());
        fUit.println("Punte : " + Deelnemer.berekenPunte());
        fUit.println("Kategorie wat meeste waargeneem is : " +
                    Deelnemer.mesteGesien());
    } ✓✓

    fUit.close();✓

    System.out.println();
    System.out.println("Resultate Suksesvol Geskryf na Lêer");
    break;
}

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

    } // switch

} // while

} // main

} // class
//=====
```

V 2.2.2

(1) Roep die toString-metode
van die objek
(1) in 'n vertoon-stelling

V 2.2.3

(1) Stel die naam van die
nuwe lêer reg saam
(1) Maak die nuwe lêer oop
(1) Stel die inligting reg saam
om na die lêer te skryf
(1) Skryf na die lêer
(1) Maak lêer toe

[49]

VRAAG 3: JAVA PROGRAMMERING

NB: Hierdie is slegs 'n voorbeeld – leerders kan hierdie vraag op enige manier antwoord soos hulle wil. Maak gebruik van die algemene rubriek wat ingesluit is by die nasienblaaie vir nasien.

ToetsAfstande.java**//Vraag 3.1 (3)**

```
import java.util.Scanner;
import java.io.*;
import javax.swing.*;
public class ToetsAfstande
{
    String ✓ [] arrData = new String [12]; ✓
    public ToetsAfstande()
    {
        arrData[0] = "12,15:02h00";
        arrData[1] = "13,10:05h00";
        arrData[2] = "9,20:06h00";
        arrData[3] = "10,15:09h00";
        arrData[4] = "7,8:10h00";
        arrData[5] = "10,10:11h0";
        arrData[6] = "12,18:14h00";
        arrData[7] = "7,18:17h00";
        arrData[8] = "11,7:19h00";
        arrData[9] = "10,10:20h00";
        arrData[10]= "2,1:23h00";
        arrData[11]= "12,17:24h00";✓
    }
}
```

V 3.1

(2) Verklaar skikking van stringe
(1) Verwyder kommentaar tekens sodat string aan skikking toegeken kan word

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

//Vraag 3.2 (17)

```
public void vertoonAfstande()
{
    int waterX = 10;
    int waterY = 10; } ✓
    int xPos;
    int yPos;
    System.out.println("Afstande van die
                        watergat af");✓
    System.out.printf("%-10s%-15s%-10s%-10s",
                      "Tyd", "Afstand(km)",
                      "X-pos", "Y-pos");✓
    System.out.println();
    int afstand;
    String tyd;
    for (int i=0;i<12;i++)✓
    {
        String reel = arrData[i];
        int komma = reel.indexOf(','); ✓
        xPos = Integer.parseInt(reel.substring
                                (0, komma)); ✓
        int dPunt = reel.indexOf(':'); ✓
        yPos = Integer.parseInt(reel.substring(komma +
                                                1,dPolon)); ✓
        tyd = reel.substring(dPunt + 1); ✓✓
    }
}
```

V 3.2

(1) Inisialiseer waterX & waterY
(2) Vertoon opskrif & subopskrifte
(1) Lus met krulhakies op regte plekke
Binne die lus:
(2) Onttrek x-pos,
(2) Onttrek y-pos,
(2) Onttrek tyd uit skikking se string
(3) Bereken afstand,
(1) Antwoord afgerond,
(3) Vertoon alle inligting

```

afstand = (int) Math.round( Math.sqrt( Math.pow((xPos- waterX),2)
+ Math.pow((yPos - waterY ),2)));

System.out.printf("%-10s%-15s%-10s%-10s",
                  tyd, afstand ,xPos,yPos);

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

```

//Vraag 3.3 (16)

```

public void etikette()
{
    System.out.print("Tik die aantal verskillende tipes diere in die
                     groep in : ");
    int n = inKb.nextInt();
    for (int i = 1; i <= n; i++)
    {
        inKb.nextLine();
        System.out.println();
        System.out.print("Tik die naam van dier
                         tipe "+ i + ": ");
        String dier = inKb.nextLine();

        System.out.print("Tik die aantal diere
                         in die groep van dier tipe "+ i + ": ");
        int aantal = inKb.nextInt();

        String etiket = dier.substring(0,2);

        etiket = etiket +
                dier.charAt(dier.length()- 1);

        int randomGetal;
        do
        {
            randomGetal = (int)(Math.random() * 900
                                + 100);

        }
        while (randomGetal % 2 !=0);

        etiket = etiket + randomGetal;
        System.out.printf("%-20s%-20s",dier,"Etiket");
        System.out.println();
        for (int j = 1; j<= aantal; j++)
        {
            System.out.printf("%-20s%-20s",j+".",etiket+"-"+j);
            System.out.println();
        }
    }
}
//=====

```

V 3.3

- (1) Tik aantal verskillende tipes diere in
- (1) Lus van 1 tot die aantal verskillende tipes diere
- (1) Binne lus:
 - (1) Tik naam van diersoort in
 - (1) Tik die aantal in die groep in van diersoort
 - (1) Onttrek eerste 2 letters en ken toe aan etiket
 - (2) Onttrek die laaste letter
 - (2) Genereer ewekansige getal in regte reeks
 - (1) Toets vir ewegetal
 - (1) Voeg random getal by etiket
 - (1) Opskrif
 - (1) Binneste lus om nommers van diere binne die groep by etiket te voeg
 - (2) Vertoon die korrekte data reg gespasieer binne die binneste lus

OF

```

randomGetal =(int)(Math.random()*899+100);
if (randomGetal % 2 != 0)
    randomGetal++;

```

```

public static void main(String[] args) throws Exception
{
    Scanner input = new Scanner(System.in);
    ToetsAfstande obj = new ToetsAfstande();
    char ch = ' ';
}

```

```

while (ch != 'V')
{
    System.out.println();
    System.out.println("      KEUSELYS");
    System.out.println(" ");
    System.out.println("      Opsie A");
    System.out.println("      Opsie B");
    System.out.println(" ");
    System.out.println("      V - VERLAAT");
    System.out.println(" ");
    System.out.print("      Jou keuse? ");

    ch = input.nextLine().toUpperCase().charAt(0);

    switch (ch)
    {
        case 'A':
        {
            obj.vertoonAfstande();
            break;
        }

        case 'B':
        {
            obj.etiketete();
            break;
        }

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

    } // switch

} // while
}
}
//=====
}

```

Alternatiewe struktuur: word nie aanbeveel nie maar moet aanvaar word:
Die kode kan gebruik word in plaas van die "case"-stelling. Sien elektroniese oplossing.

[36]**EINDE VAN AFDELING B: JAVA****TOTAAL AFDELING B: 120****GROOTTOTAAL: 120**

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

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG 1: DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS PUNTE	LEERDER SE PUNTE
1.1	SELECT *✓ FROM tblWaarnemings✓ ORDER BY✓ WaarneemID DESC✓	4	
1.2	SELECT DISTINCT✓ Dier✓FROM tblWaarnemings✓ WHERE Jonk = true ✓	4	
	SELECT DISTINCT✓ Dier✓FROM tblWaarnemings✓ WHERE Jonk = Yes ✓		
	SELECT DISTINCT✓ Dier✓FROM tblWaarnemings✓ WHERE Jonk = On ✓		
1.3	SELECT VeldwagterID, Naam, Van, ✓ year(Now())✓ - ✓year (DatumAangestel)AS [TotaleJare] ✓FROM tblVeldwagters	6	
	SELECT VeldwagterID, Naam, Van, ✓ year(Now())✓ - ✓year (DatumAangestel)AS TotaleJare ✓FROM tblVeldwagters		
	SELECT VeldwagterID, Naam, Van, ✓ 2010✓ - ✓ year (DatumAangestel)AS [TotaleJare] ✓FROM tblVeldwagters		
	SELECT VeldwagterID, Naam, Van, ✓ year(Date())✓ - ✓year (DatumAangestel)AS [TotaleJare] ✓FROM tblVeldwagters		
1.4	SELECT Dier✓, format(Avg(AantalDiere) ✓,"0.00"✓) AS [GemWaarnemings] ✓FROM tblWaarnemings✓GROUP BY Dier✓	6	
	SELECT Dier✓, round(Avg(AantalDiere) ✓,2✓) AS [GemWaarnemings] ✓FROM tblWaarnemings✓GROUP BY Dier✓		
1.5	Toevoer-stelling✓ 'DELETE✓ FROM tblWaarnemings✓ WHERE WaarneemID = ' + IntToStr(id); ✓	4	
1.6	UPDATE✓ tblWaarnemings✓ SET✓ Dier = "Witrenoster"✓ WHERE Dier = "Renoster"	5	
	UPDATE✓ tblWaarnemings✓ SET✓ Dier = "Witrenoster"✓ WHERE Dier Like "%Renoster%"		
1.7	SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings, tblVeldwagters ✓ WHERE tblWaarnemings.VeldwagterID = ✓ tblVeldwagters.VeldwagterID✓ AND Dier = "Olifant"✓ AND WaarneemDatum > #30/04/2010#✓	6	
	SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings, tblVeldwagters ✓ WHERE tblWaarnemings.VeldwagterID = ✓ tblVeldwagters.VeldwagterID✓ AND Dier = "Olifant"✓ AND month(WaarneemDatum) > 4✓		
	SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings, tblVeldwagters ✓ WHERE tblWaarnemings.VeldwagterID = ✓ tblVeldwagters.VeldwagterID✓ AND Dier LIKE "Olifant"✓ AND WaarneemDatum > #30/04/2010#✓		
	TOTAAL:	35	

ADDENDUM C**VRAAG 3: DELPHI PROGRAMMERING**

SENTRUMNOMMER:.....		EKSAMENNOMMER:	
VRAAG 3 DELPHI – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
3.1	(2) Verklaar arrData-skikking (1) Verwyder kommentaar-tekens om stringe toe te ken aan skikking	3	
3.2	Opsie A (1) Ken 10 toe aan waterX en waterY (2) Opskrif en subopskrifte (1) Lus deur die gegewe skikking Binne Lus: (2) Onttrek xPos en (2) yPos (2) Onttrek tyd (3) Bereken afstand, (1) rond afwaarts af (3) Vertoon info in lus	17	
3.3	Opsie B (1) Tik die aantal verskillende tipes diere in, (1) Lus van 1 tot die aantal verskillende tipes diere, (1) Binne die lus: (1) Tik tipe dier in, (1) Tik die hoeveelheid van die dier in die groep in, (1) Onttrek eerste 2 letters en ken toe aan etiket (2) Onttrek die laaste letter, (2) Genereer getal in regte reeks, (1) Toets vir ewe getal, (1) Voeg random getal en koppelteken by die etiket (1) Opskrif (1) Binneste lus vir tipe dier Voeg 'n unieke nommer by die etiket vir elke dier van een soort in die groep, (2) Vertoon die regte data reg gespasieer in die binneste lus	16	
	TOTAAL:	36	

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

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG 1: JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS PUNTE	LEERDER SE PUNTE
1.1	SELECT *✓ FROM tblWaarnemings✓ ORDER BY✓ WaarneemID DESC✓	4	
1.2	SELECT DISTINCT✓ Dier✓FROM tblWaarnemings✓ WHERE Jonk = true ✓	4	
	SELECT DISTINCT✓ Dier✓FROM tblWaarnemings✓ WHERE Jonk = yes ✓		
	SELECT DISTINCT✓ Dier✓FROM tblWaarnemings✓ WHERE Jonk = 1 ✓		
1.3	SELECT VeldwagterID, Naam, Van, ✓ year(Now())✓ - ✓year (DatumAangestel)AS [TotaleJare] ✓FROM tblVeldwagters	6	
	SELECT VeldwagterID, Naam, Van, ✓ year(Now())✓ - ✓year (DatumAangestel)AS TotaleJare ✓FROM tblVeldwagters		
	SELECT VeldwagterID, Naam, Van, ✓ 2010✓ - ✓ year (DatumAangestel)AS [TotaleJare] ✓FROM tblVeldwagters		
	SELECT VeldwagterID, Naam, Van, ✓ year(Date())✓ - ✓year (DatumAangestel)AS [TotaleJare] ✓FROM tblVeldwagters		
1.4	SELECT Dier✓, format(Avg(AantalDiere) ✓,"0.00"✓) AS [GemWaarnemings] ✓FROM tblWaarnemings✓GROUP BY Dier✓	6	
	SELECT Dier✓, round(Avg(AantalDiere) ✓,2✓) AS [GemWaarnemings] ✓FROM tblWaarnemings✓GROUP BY Dier✓		
1.5	Toevoer-stelling✓ "DELETE✓ FROM tblWaarnemings✓ WHERE WaarneemID = " + id; ✓ // of "+id+""	4	
1.6	UPDATE✓ tblWaarnemings✓ SET✓ Dier = 'Witrenoster'✓ WHERE Dier = 'Renoster'	5	
	UPDATE✓ tblWaarnemings✓ SET✓ Dier = 'Witrenoster'✓ WHERE Dier Like '%Renoster%'		
1.7	SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings, tblVeldwagters ✓ WHERE tblWaarnemings.VeldwagterID = ✓ tblVeldwagters.VeldwagterID✓ AND Dier = 'Olifant'✓ AND WaarneemDatum > #30/04/2010#✓	6	
	SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings, tblVeldwagters ✓ WHERE tblWaarnemings.VeldwagterID = ✓ tblVeldwagters.VeldwagterID✓ AND Dier = 'Olifant'✓ AND month(WaarneemDatum) > 4✓		
	SELECT WaarneemDatum,Naam,Van✓ FROM tblWaarnemings, tblVeldwagters ✓ WHERE tblWaarnemings.VeldwagterID = ✓ tblVeldwagters.VeldwagterID✓ AND Dier LIKE 'Olifant'✓ AND WaarneemDatum > #30/04/2010#✓		
	TOTAAL:	35	

ADDENDUM E

VRAAG 2: JAVA – OBJEK-GEÖRIËNTEERDE PROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG 2 JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
2.1			
2.1.1	Definieer die attribute van Deelnemer: vier private (1) velde, drie heelgetalle met die regte name(1) en een string (1)	3	
2.1.2	Konstruktor: naam parameter (1) inisialiseer al die 'instance fields' (1) Verwyder kommentaar tekens uit die 'sien'-metodes(1)	3	
2.1.3	berekenPunte: (3) vermenigvuldig tellers met regte waardes en tel resultate bymekaar	3	
2.1.4	aantalDiere: stuur heelgetal terug (1) wat die sum van die punte bevat(1)	2	
2.1.5	getNaam metode: Regte naam en terugstuur-tipe (1), Stuur regte veld terug (1)	2	
2.1.6	MeesteGesien metode: Stuur String terug (1) twee if .. met else-stelling om hoogste teller te bepaal (3)	4	
2.1.7	toString: naam en nuwe lyn (1) en voeg string by(1) met kategorie-punte(1)voeg laaste lyn by(1) met tab (1)	5	
2.2			
2.2.1	Verklaar 'n enkele Deelnemer-objek (2) Skep 'n nuwe File-objek(1) As die lêer nie bestaan(1)vertoon boodskap en verlaat (1) as lêer bestaan, maak lêer oop om te lees (2) Lees eerste reel uit buite lus uit lêer (1) ken toe aan naam veranderlike (1) Roep konstruktor metode van Deelnemer (1) gebruik naam as parameter (1) inisialiseer teller na nul(1)lus met krulhakies op regte plekke (1) Binne die lus: Lees lyn uit lêer (1), kry dier se naam(1) en kategorie-letter (1) uit die string. Gebruik letter en vergelyk (1) roep gepaste metode om regte teller te vermeerder (3) om geldige kategoriee te hanteer. Ander vermeerder ongeldige teller en vertoon dier se naam en boodskap dat kategorie ongeldig is(1). Vertoon aantal geldig en ongeldige inskrywings(1)	20	
2.2.2	Roep toString-metode van Deelnemer-objek (1)om inligting te vertoon(1)	2	
2.2.3	Gebruik die naam van Deelnemer om lêernaam saam te stel (1) skep File-objek om nuwe lêer te skep om na lêer te skryf(1) Roep metode om afvoer saam te stel (1) en skryf na lêer (1), maak lêer toe (1)	5	
	TOTAAL:	49	

ADDENDUM F**VRAAG 3: JAVA PROGRAMMERING**

SENTRUMNOMMER:.....		EKSAMENNOMMER:	
VRAAG 3 JAVA – NASIENBLAD			
VRAAG	ASPEK	MAKS. PUNTE	LEERDER SE PUNTE
3.1	(2) Verklaare arrData-skikking (1) Verwyder kommentaar-tekens om stringe toe te ken aan skikking	3	
3.2	Opsie A (1) Ken 10 toe aan waterX en waterY (2) Opskrif en subopskrifte (1) Lus deur die gegewe skikking Binne Lus: (2) Onttrek xPos en (2) yPos (2) Onttrek tyd (3) Bereken afstand, (1) rond afwaarts af (3) Vertoon info in lus	17	
3.3	Opsie B (1) Tik die aantal verskillende tipes diere in (1) Lus van 1 tot die aantal verskillende tipes diere Binne die lus: (1) Tik tipe dier in (1) Tik die hoeveelheid van die dier in die groep in (1) Onttrek eerste 2 letters en ken toe aan etiket (2) Onttrek die laaste letter (2) Genereer getal in regte reeks (2) Toets vir ewe getal (1) Voeg random getal en koppelteken by die etiket (1) Opskrif (1) Binneste lus vir tipe dier Voeg 'n unieke nommer by die etiket vir elke dier van een soort in die groep (2) Vertoon die regte data reg gespasieer in die binneste lus	16	
	TOTAAL:	36	