



NASIONALE SENIOR CERTIFIKAAT-EKSAMEN  
NOVEMBER 2022

**INLIGTINGSTEGNOLOGIE: VRAESTEL I**  
**NASIENRIGLYNE**

Tyd: 3 uur

150 punte

---

Hierdie nasienriglyne word voorberei vir gebruik deur eksaminatore en hulpeksaminatore. Daar word van alle nasieners vereis om 'n standaardiseringsvergadering by te woon om te verseker dat die nasienriglyne konsekwent vertolk en toegepas word tydens die nasien van kandidate se skrifte.

Die IEB sal geen gesprek aanknoop of korrespondensie voer oor enige nasienriglyne nie. Daar word toegegee dat verskillende menings rondom sake van beklemtoning of detail in sodanige riglyne mag voorkom. Dit is ook voor die hand liggend dat, sonder die voordeel van bywoning van 'n standaardiseringsvergadering, daar verskillende vertolkings mag wees oor die toepassing van die nasienriglyne.

---

**AFDELING A      SQL****VRAAG 1.1 [4]**

```
SELECT *  
FROM tblKapteins  
WHERE GevegOpleiding = TRUE  
ORDER BY DatumAangestel DESC
```

**VRAAG 1.2 [4]**

```
SELECT *  
FROM tblRuimteskepe  
WHERE (Model LIKE 'M*' OR Model LIKE 'T*') AND BrandstofVerbruik  
< 500
```

**Aanvaar % vir MySQL/JavaDB**

**VRAAG 1.3 [5]**

```
SELECT *  
FROM tblKapteins  
WHERE YEAR(NOW()) - YEAR(DatumAangestel) >= 10
```

**JAVADB**

```
SELECT *  
FROM tblKapteins  
WHERE YEAR(CURRENT_DATE) - YEAR(DatumAangestel) >= 10
```

**MYSQL**

```
SELECT *  
FROM tblKapteins  
WHERE YEAR(NOW() ) - YEAR(DatumAangestel) >= 10
```

**VRAAG 1.4 [5]**

```
SELECT Model, Spoed  
FROM tblRuimteskepe  
WHERE Spoed > (SELECT AVG(Spoed) FROM tblRuimteskepe)  
ORDER BY Spoed DESC
```

**VRAAG 1.5 [7]**

```
SELECT YEAR(VertrekDatum) AS JaarVanSending, COUNT(*) AS  
GetalSendings  
FROM tblSendings  
WHERE MONTH(VertrekDatum) <= 6  
GROUP BY YEAR(VertrekDatum)
```

**MySQL Alias use in GroupBy**

```
SELECT YEAR(VertrekDatum) AS JaarVanSending, COUNT(*) AS  
GetalSendings  
FROM tblSendings  
WHERE MONTH(VertrekDatum) <= 6  
GROUP BY JaarVanSending
```

**VRAAG 1.6 [9]**

```
SELECT tblSendings.SendingID, tblKapteins.VolleNaam,  
tblRuimteskepe.Model, tblRuimteskepe.Spoed,  
tblSendings.Sonnestelsel, tblSendings.Afstand, Round  
( (Afstand/Spoed),2) AS TydInJare  
FROM tblSendings, tblKapteins, tblRuimteskepe  
WHERE tblSendings.KapteinID=tblKapteins.KapteinID AND  
tblSendings.RuimteskipID=tblRuimteskepe.RuimteskipID;
```

**JAVADB**

```
SELECT tblSendings.SendingID, tblKapteins.VolleNaam,  
tblRuimteskepe.Model, tblRuimteskepe.Spoed,  
tblSendings.Sonnestelsel, tblSendings.Afstand, FLOOR (  
(Afstand/Spoed) * 100 + 0.5) / 100 AS TydInJare  
FROM tblSendings, tblKapteins, tblRuimteskepe  
WHERE tblSendings.KapteinID=tblKapteins.KapteinID AND  
tblSendings.RuimteskipID=tblRuimteskepe.RuimteskipID
```

**MYSQL**

```
SELECT tblSendings.SendingID, tblKapteins.VolleNaam,  
tblRuimteskepe.Model, tblRuimteskepe.Spoed,  
tblSendings.Sonnestelsel, tblSendings.Afstand, Round  
( (Afstand/Speed),2) AS TydInJar  
FROM tblSendings, tblKapteins, tblRuimteskepe  
WHERE tblSendings.KapteinID=tblKapteins.KapteinID AND  
tblSendings.RuimteskipID=tblRuimteskepe.RuimteskipID;
```

**Aanvaar Inner Joins****VRAAG 1.7 [7]**

```
UPDATE tblSendings  
SET Hoofdoelwit = 'Scouting'  
WHERE VertrekDatum > NOW() AND RIGHT(Sonnestelsel,1) NOT IN  
('A','B','C') AND Hoofdoelwit = 'Military Exercise'
```

**JAVAD**

```
UPDATE tblSendings  
SET Hoofdoelwit = 'Scouting'  
WHERE VertrekDatum > CURRENT_DATE AND SUBSTR(Sonnestelsel,  
length(Sonnestelsel) , 1) NOT IN ('A' , 'B' , 'C') AND  
Hoofdoelwit = 'Military Exercise'
```

**MYSQL**

```
UPDATE tblSendings
SET Hoofdoelwit = 'Scouting'
WHERE VertrekDatum > CURDATE() AND RIGHT(Sonnestelsel,1) NOT IN
('A','B','C') AND Hoofdoelwit = 'Military Exercise'
```

**VRAAG 1.8 [9]**

```
INSERT INTO tblSendings (KapteinID, RuimteskipID, Hoofdoelwit,
VertrekDatum, Afstand, Sonnestelsel)
SELECT KapteinID , INT( (RND(KapteinID) * 30) + 1 ) ,
'Exploration' , #01/01/2030# , 5 , 'Alpha-Centauri'
FROM tblKapteins
WHERE KapteinID NOT IN (SELECT KapteinID FROM tblSendings)
```

**MYSQL**

```
INSERT INTO tblSendings ( KapteinID, RuimteskipID, Hoofdoelwit,
VertrekDatum, Afstand, Sonnestelsel )
SELECT KapteinID, FLOOR( (RAND() * 30) + 1 ), 'Exploration',
'01/01/2030', 5, 'Alpha-Centauri'
FROM tblKapteins
WHERE KapteinID NOT IN (SELECT KapteinID FROM tblSendings)
```

**JAVADB**

```
INSERT INTO tblSendings ( KapteinID, RuimteskipID, Hoofdoelwit,
VertrekDatum, Afstand, Sonnestelsel )
SELECT KapteinID, FLOOR( (RAND(KapteinID) * 30) + 1 ) ,
'Exploration', '01/01/2030' , 5, 'Alpha-Centauri'
FROM tblKapteins
WHERE KapteinID NOT IN (SELECT KapteinID FROM tblSendings)
```

**AFDELING B      OBJEKGEORIËNTEERDE PROGRAMMERING****JAVA-OPLOSSING****VRAAG 2      BemanningLid.java**

```
//V2.1 - 4
Klasopskrif korrek
public class BemanningLid {
    velde privaat
    korrek getipeer
    korrek benoem
    private String vollenaam;
    private int bemanningID;
    private String departement;

//V2.2 - 2
veld as openbaar verklaar
korrek benoem en toegeken (moet staties wees)
public static int getalBevorderdeBemanning=0;

//V2.3 - 3
opskrif korrek
parameters korrek benoem en getipeer
public BemanningLid(String inVN , int inBID, String inDT )
{
    velde toegeken deur parameters te gebruik
    vollenaam = inVN;
    bemanningID = inBID;
    departement = inDT;
}

//V2.4 - 3
Alle krymetodes korrek benoem
Alle terugsendings korrekte tipe verklaar
Alle terugsendings korrekte veld
public String kryVolleNaam() {
    return vollenaam;
}

public int kryBemanningID() {
    return bemanningID;
}

public String kryDepartement() {
    return departement;
}
```

```

//V2.5 - 5
Opskrif korrek
public String naString()
{
    bevat alle velde
    tabuleerspasies bygevoeg
    formatering korrek
    stuur string terug
    return vollenaam + "\tCrew ID: " + bemanningID + "\t[" +
departement + "]\n";
}

}

```

### VRAAG 3    Offisier.java

```

//V3.1 - 2
Klas korrek benoem
Brei BemanningLid uit
public class Offisier extends BemanningLid
{
    //V3.2 - 2
    privaat, korrek benoem en getipeer
    datumBevorder gebruik 'n geskikte datumobjek
    private int rang;
    private LocalDate datumBevorder;
    //V3.3 - 2
    veld as openbaar verklaar
    korrek benoem en toegeken (moet staties wees)
    public static int getalBevorderdeOffisiere = 0;

    //V3.4 - 4
    Konstruktoropskrif korrek

    public Offisier(String inVN , int inBID, String inDT ,int
inRG, LocalDate inDB)
    {
        Ouerkonstruktor geroep
        Korrekte parameters gegee
        super(inVN, inBID, inDT);

        dogterklasvelde toegeken deur parameters te gebruik
        rang = inRG;
        datumBevorder = inDB;
    }

    //V3.5 - 1
    Metodeopskrif korrek en stuur korrekte veld terug

    public int kryRang() {
        return rang;
    }
}

```

```
//V3.6 - 5
Metodeopskrif korrek
public String kryTitel()
{
    skikking van stringe wat verskillende titels
    in korrekte volgorde verteenwoordig
    String[] titelSkik =
    {
        "", "Ensign", "Lieutenant", "Lt Commander",
"Commander", "Captain"
    };

    stuur korrekte titel terug op grond van rang
    return titelSkik[rang];
}

//V3.6 - 5
**Alternatiewe oplossing**
**Switch kan vervang word deur case-stelling of korrek
geneste if else-stelling
Opskrif korrek
public String kryTitel()
{
    kontroleer titel op grond van rang
    String titel = "";
    switch(rang)
    {
        case 1:
            titel = "Ensign";
            break;
        case 2:
            titel = "Lieutenant";
            break;
        case 3:
            titel = "Lt Commander";
            break;
        case 4:
            titel = "Commander";
            break;
        case 5:
            titel = "Captain";

    }
    stuur korrekte titel terug
    return titel;
}
```

```
//V3.7 - 4
Metodeopskrif korrek
public String naString()
{
    roep naString van ouerklas
    las titel aan
    stuur string terug
    return super.naString() + " " + kryTitel();
}

//V3.8 - 5
Metodeopskrif korrek
public void bevorder()
{
    Kontroleer of rang kleiner is as 5
    (kleiner as of gelyk aan 4 ook aanvaarbaar)
    if (rang < 5)
    {
        inkrementeer rang
        rang = rang + 1;
        inkrementeer getal bevorderde offisiere
        getalBevorderdeOffisiere++;

        Bywerking van die bevorderingsdatum tot die huidige datum
        datumBevorder = LocalDate.now();
    }
}

}
```

#### **VRAAG 4 & 6.1, 6.2    BemanningLidBestuurder.java**

```
//V4.1 - 1
Klasopskrif korrek
public class BemanningLidBestuurder
{
    //V4.2 - 4
    velde privaat
    velde korrek benoem
    skikking van 80 BemanningLid-objekte (Tipe van die ouerklas)
    grootte-veld korrekte tipe

    private BemanningLid[] bSkik = new BemanningLid[80];
    private int grootte = 0;
```



```

//V4.3 - 11
Konstruktormetodeopskrif korrek
public BemanningLidBestuurder()
{
    try
    {
        maak lêer oop
        Scanner sc = new Scanner(new
File("bemanningslede.txt"));

        lus deur lêer
        while (sc.hasNextLine())
        {
            lees reël
            String line = sc.nextLine();

            onttrek eerste drie velde
            String tokens[] = line.split("#");
            String vollenaam = tokens[0];
            int bemanningID = Integer.parseInt(tokens[1]);
            String departement = tokens[2];

            kontroleer vir 'n offisier of 'n bemanningslid
            if (tokens.length == 3)
            {
                skep BemanningLid-objek en voeg by skikking by
                BemanningLid c = new BemanningLid(vollenaam,
bemanningID, departement);
                bSkik[grootte] = c;
            }
            else
            {
                kry offisierrang
                int rang = Integer.parseInt(tokens[3]);
                kry datum en skakel datum om (datum moet ooreenstem
                met die objektipe in Offisier gedefinieer)
                LocalDate datum = LocalDate.parse(tokens[4],
DateTimeFormatter.ofPattern("dd/MM/yyyy"));
                skep Offisier-objek en voeg by skikking by
                Offisier f = new Offisier(vollenaam,
bemanningID, departement, rang, datum);
                bSkik[grootte] = f;
            }
            inkrementeer grootte
            grootte++;

        }

        sc.close();
    } catch (FileNotFoundException fne)
    {
        System.out.println("File Missing " + fne);
    }
}

```

```
//V4.4 - 5
metodeopskrif korrek
public String naString()
{
    String r = "";
    lus deur skikking
    for (int i = 0; i < grootte; i++)
    {
        las by string aan
        voeg nuwe reël by
        r += bSkik[i].naString() + "\n";
    }

    stuur aangelaste string terug
    return r;
}

//V6.1 - 4
soek die bemanningslid deur bemanningID te gebruik
public int soekBemanningLid(int bemanningID)
{
    int pos = 0;
    gebruik 'n toepaslike lus
    while (!found && pos < grootte)
    {
        beëindig soektog indien bemanningID ooreenstem
        if (bSkik[pos].kryBemanningID() == bemanningID)
        {
            stuur pos terug;
            pos++;
        }

        stuur terug -1 not found
        return -1;
    }
}

//V6.2 - 5
Metodeopskrif korrek met return-stelling
public String verwerkToetsUitslae()
{
    lys vir afvoer
    String r = "";
    try
    {
        lees lêer
        Scanner sc = new Scanner(new
File("toetsUitslae.txt"));
        loop deur deur toepaslike lus te gebruik
        for (int i = 0; i < 10; i++)
        {
            lees die bemanningID en toetsUitslag
            int bemanningID = sc.nextInt();
            int toetsUitslag = sc.nextInt();
        }
    }
}
```

```
//V6.2.1 - 1
soek die pos van die bemanningslid met die
bemanningID
int pos = soekBemanningLid(bemanningID);

//V6.2.2a - 7
kontroleer of die bemanningslid 'n offisier is
if (bSkik[pos] instanceof Offisier)
{
    skakel element om in Offisier
    Offisier f = (Offisier) bSkik[pos];
    boolean slaag = false;
    kontroleer of die offisier aan die vereistes
    voldoen het
    korrek geneste if-stelling
    if (f.kryRang() == 1 && toetsUitslag >= 80)
    {
        slaag = true;
    } else if (f.kryRang() == 2 && toetsUitslag >=
80)
    {
        slaag = true;
    } else if (f.kryRang() == 3 && toetsUitslag >=
85)
    {
        slaag = true;
    } else if (f.kryRang() == 4 && toetsUitslag >=
90)
    {
        slaag = true;
    }
}

//V6.2.3 - 6
6.2.3 - las die woord "Officer" en die naam aan
r += "Officer " + f.kryVolleNaam();

//6.2.2a vervolg
kontroleer of hy/sy geslaag het
if (slaag)
{
    bevorder die offisier
    f.bevorder();
    vervang objek in die skikking met die
    nuwe objek
    bSkik[pos] = f;

    6.2.3 - Las slaagteks vir offisier aan
    r += " has passed and is promoted to " +
        f.kryTitel() + " Rank " +
        f.kryRang() + "\n";
}
else if (f.kryRang() == 5)
{
```

```

        6.2.3 - las kapteindeelname aan
        r += " participated in the test\n";
    }
    else if (slaag == false)
    {
        6.2.3 - las druip vir 'n offisier aan
        r += " has failed\n";
    }
}
//6.2.2b - 6
else
{
    r += "Crew Member " +
bSkik[pos].kryVolleNaam();
    kontroleer of die bemanningslid die minimum
    behaal het om te slaag en las slaag/druip aan
    if (toetsUitslag >= 75)
    {
        inkrementeer getalBevorderdeBemanning
        BemanningLid.getalBevorderdeBemanning++;
        bevorder die bemanningslid deur nuwe
        offisier met bemanningslidbesonderhede te skep
        deur toepaslike datum te gebruik
        Offisier nuweOffisier = new Offisier
            (bSkik[pos].kryVolleNaam(),
            bSkik[pos].kryBemanningID(),
            bSkik[pos].kryDepartement(),
            1, LocalDate.now());
        voeg nuwe Offisier-objek by die skikking by
        bSkik[pos] = nuweOffisier;
        6.2.3 - Las slaagboodskap by nuwe offisier
        aan en sluit nuwe titel en rang in
        r += " has passed and is promoted to " +
            nuweOffisier.kryTitel() + " Rank " +
            nuweOffisier.kryRang() + "\n";
    } else
    {
        6.2.3 - Las druip vir 'n bemanningslid aan
        r += " has failed\n";
    }
}

}
} catch (FileNotFoundException e)
{
    System.out.println("File Missing " + e);
}

return r;
}
}

```

**VRAAG 5, 6.3, 6.4      BemanningLidGK.java**

```
//V5.1 - 1
    Klasopskrif korrek
public class BemanningLidGK {
    public static void main(String[] args) {
        //V5.2 - 2
        BemanningLidBestuurder-objek op toepaslike plek geskep
        BemanningLidBestuurder cm = new BemanningLidBestuurder();

        //V5.3 - 2
        Roep naString en vertoon alle bemanningslede
        System.out.println(cm.naString());

        //V6.3 - 1
        Roep verwerkToetsUitslae in toepaslike klas

        System.out.println(cm.verwerkToetsUitslae());

        //V6.4 - 2
        Roep die statiese getalBevorderdeBemanning
        Roep die statiese getalBevorderdeOffisiere
        System.out.println("Number of promoted crew members: " +
                           BemanningLid.getalBevorderdeBemanning);
        System.out.println("Number of promoted officers: " +
                           Offisier.getalBevorderdeOffisiere);
    }
}
```

**DELPHI-OPLOSSING****VRAAG 2    uBemanningLid.pas**

```
unit uBemanningLid;

interface
uses SysUtils;

//V2.1 - 4
  Klasopskrif korrek
type TBemanningLid = class
  velde privaat
  korrek getipeer
  korrek benoem
private
  vollenaam : string;
  bemanningID : integer;
  departement : string;

public
  //V2.2 - 2
  veld as openbaar verklaar
  korrek benoem en toegeken (moet staties/klas wees)
  class var getalBevorderdeBemanning : integer;
  constructor Create(inVN: string; inBID : integer; inDT :
string );
  function kryVolleNaam() : string;
  function kryBemanningID() : integer;
  function kryDepartement() : string;
  function naString() : string ; virtual;

end;

implementation

{ TBemanningLid }

//V2.3 - 3
  Konstruktoropskrif korrek
  Parameters korrek benoem en getipeer
constructor TBemanningLid.Create(inVN: string; inBID: integer;
inDT: string);
begin
  velde toegeken deur parameters te gebruik
  vollenaam := inVN;
  bemanningID := inBID;
  departement := inDT;
end;
```

```
//V2.4 - 3
Alle krymetodes korrek benoem
Alle metodeterugstuurtypes in opskrif korrek
Alle metodes stuur korrekte veld terug
function TBemanningLid.kryBemanningID: integer;
begin
    Result := bemanningID;
end;

function TBemanningLid.kryDepartement: string;
begin
    Result := departement;
end;

function TBemanningLid.kryVolleNaam: string;
begin
    Result := vollenaam;
end;

//V2.5 - 5
Metodeopskrif korrek
function TBemanningLid.naString: string;
begin
    bevat alle velde
    tabuleerspasië bygevoeg
    formatering korrek
    stuur string terug
    Result := vollenaam + #9 + 'CrewID: ' + IntToStr(bemanningID) +
#9 + '[' + departement + ']';
end;

end.
```

### VRAAG 3 uOffisier.pas

```
unit uOffisier;

interface
    uses SysUtils, DateUtils, uBemanningLid;
//V3.1 - 2
Klas korrek benoem
Brei BemanningLid uit

type TOffisier = class(TBemanningLid)

    //V3.2 - 2
    privaas, korrek benoem en getipeer
    datumBevorder gebruik 'n geskikte datumobjek
private
    rang : integer;
    datumBevorder : TDateTime;

public
    //V3.3 - 2
```

```

    veld as openbaar verklaar
    korrek benoem en toegeken (staties/klas)

    class var getalBevorderdeOffisiere : integer;
    constructor Create(inVN: string; inBID : integer; inDT :
string; inRG : integer; inDB : TDateTime);
    function kryTitel() : string;
    function kryRang() : integer;
    function naString : string ; override;
    procedure bevorder();
end;

implementation
    { TOffisier }
//V3.4 - 4
Konstruktoropskrif korrek
constructor TOffisier.Create(inVN: string; inBID : integer; inDT :
string; inRG : integer; inDB : TDateTime);
begin
    Ouerkonstruktor geroep
    Korrekte parameters gegee
    Inherited Create(inVN, inBID, inDT);
    dogterklasvelde toegeken deur parameters te gebruik
    rang := inRG;
    datumBevorder := inDB;
end;

//V3.5 - 1
Metodeopskrif korrek en stuur korrekte veld terug
function TOffisier.kryRang: integer;
begin
    Result := rang;
end;

//V3.6 - 4
Metodeopskrif korrek
function TOffisier.kryTitel: string;
const
    skikking van stringe wat verskillende titels in korrekte
    volgorde verteenwoordig
    titelSkik : array[1..5] of string = ('Ensign','Lieutenant','Lt
Commander','Commander','Captain');
begin
    stuur korrekte titel terug op grond van rang
    Result:= titelSkik[rang];
end;

//V3.6 - 4
**Alternatiewe oplossing**
**Switch kan vervang word deur case-stelling of korrek geneste if
else-stelling
Opskrif korrek
function TOffisier.kryTitel: string;
var

```



```
    titel : string;
begin
    kontroleer titel op grond van rang
    case rang of
        1 : titel:= 'Ensign';
        2 : titel:= 'Lieutenant';
        3 : titel:= 'Lt Commander';
        4 : titel:= 'Commander';
        5 : titel:= 'Captain';

    end;

    stuur korrekte titel terug op grond van rang
    Result:= titel;
end;
}

//V3.7 - 4
Metodeopskrif korrek
function TOffisier.naString: string;
begin
    roep naString van ouerklas
    las titel aan
    stuur string terug
    Result := Inherited naString + ' ' + kryTitel();
end;

//V3.8 - 5
Metodeopskrif korrek
procedure TOffisier.bevorder();
begin
    Kontroleer of rang kleiner is as 5 (kleiner as of gelyk aan 4 ook
aanvaarbaar)
    if rang < 5 then
    begin
        inkrementeer rang
        rang := rang + 1;
        inkrementeer getal bevorderde offisiere
        Inc(getalBevorderdeOffisiere);
    end;
    Bywerking van die bevorderingsdatum tot die huidige datum
    datumBevorder:= Now;

    end;

end;

end.
```

**VRAAG 4 & 6.1, 6.2      uBemanningLidBestuurder.pas**

```
unit uBemanningLidBestuurder;

interface
uses SysUtils, DateUtils, uBemanningLid, uOffisier;
//V4.1 - 1
Klasopskrif korrek

type tBemanningLidBestuurder = class
    //V4.2 - 4
    Velde privaat
    Velde korrek benoem
    Skikking van 80 BemanningLid-objekte (Tipe van die ouerklas)
    grootte-veld korrekte tipe
private
    bSkik : array[1..80] of TBemanningLid;
    grootte : integer;
public
    constructor Create();
    function naString() : string;
    function verwerkToetsUitslae() : string;
    function soekBemanningLid(inBID : integer) : integer;

end;
implementation

{ tBemanningLidBestuurder }

//V4.3 - 11
Konstruktormetodeopskrif korrek
constructor tBemanningLidBestuurder.Create;
var
    inFile : textfile;
    line, vollenaam, departement, datum, d, m, y: string;
    bemanningID, rang : integer;
    datumBevorder : TDateTime;
begin
    if FileExists('bemanningslede.txt') <> true then
        begin
            WriteLn('File Missing');
        end
    else
        begin
            maak lêer oop
            AssignFile(inFile, 'bemanningslede.txt');
            Reset(inFile);

            grootte := 0;
            lus deur lêer
            while NOT EOF(inFile) do
                begin
                    lees reël
                    ReadLN(inFile, line);
```

```

    inkrementeer grootte
    Inc(grootte);

    onttrek die eerste drie velde
    vollenaam := Copy(line , 1 , Pos('#', line) -1);
    Delete(line, 1, Pos('#', line));
    bemanningID := StrToInt(Copy(line , 1 , Pos('#', line) -
1));
    Delete(line, 1, Pos('#', line));

    kontroleer vir 'n offisier of 'n bemanningslid
    if Pos('#', line) > 0 then
        begin
            kry offisier se titel en rang
            departement := Copy(line , 1 ,
                Pos('#', line) -1);
            Delete(line, 1, Pos('#', line));
            rang := StrToInt(Copy(line , 1 ,
                Pos('#', line) -1));
            Delete(line, 1, Pos('#', line));
            datum := line;

            d := Copy(datum,1 , Pos('/', datum) - 1);
            Delete(datum,1 , Pos('/', datum));

            m := Copy(datum,1 , Pos('/', datum) - 1);
            Delete(datum,1 , Pos('/', datum));
            y := datum;
            kry datum (datum moet ooreenstem met die
            objektipe in Offisier gedefinieer)

            datumBevorder := EncodeDate(StrToInt(y),
                StrToInt(m),
                StrToInt(d));

            skep offisierobjek en voeg by skikking by
            bSkik[grootte] := TOffisier.Create(vollenaam,
                bemanningID, departement,
                rang, datumBevorder);

        end
    else
        begin
            departement := line;
            skep BemanningLid-objek en voeg by skikking by
            bSkik[grootte] := TBemanningLid.Create(vollenaam,
                bemanningID,
                departement);

        end;
    end;
end;
end;
end;
end;

```

```
//V4.4 - 5
```

```
Metodeopskrif korrek
```

```
function tBemanningLidBestuurder.naString: string;
var
  i : integer;
  output : string;
begin
  output := '';
  lus deur skikking
  for i := 1 to grootte do
    begin
      las by string aan
      voeg nuwe reël by
      output := output + bSkik[i].naString() + #13#10;
    end;
  stuur aangelaste string terug
  Result := output;
end;
```

```
//V6.1 - 4
```

```
soek die bemanningslid deur bemanningID te gebruik
```

```
function tBemanningLidBestuurder.soekBemanningLid(inBID: integer):
integer;
var
  found : boolean;
  pos : integer;
begin

  found := false;
  pos := 1;
  gebruik 'n toepaslike lus
  while NOT(found) AND (pos <= grootte) do
    begin
      beëindig soektog indien bemanningID ooreenstem
      Inc (pos)
      if bSkik[pos].kryBemanningID() = inBID then
        begin
          found := true;
        end;
    end;

    stuur pos terug
  Result := pos;
end;
```

```
//V6.2 - 5
Metodeopskrif korrek
function tBemanningLidBestuurder.verwerkToetsUitslae: string;
var
    infile : textfile;
    lys vir afvoer
    cLine, tLine, output : string;
    bemanningID, toetsUitslag, pos , i : integer;
    f : TOffisier;
    slaag : boolean;
begin
    lees lêer
    if FileExists('toetsUitslae.txt') <> true then
        begin
            WriteLn('File Missing');
        end
    else
        begin

            AssignFile(inFile, 'toetsUitslae.txt');
            Reset(inFile);
            loop deur deur toepaslike lus te gebruik
            for i := 1 to 10 do
                begin
                    lees die bemanningID en toetsUitslag
                    ReadLn(infile , cLine);
                    ReadLn(infile , tLine);

                    bemanningID := StrToInt(cLine);
                    toetsUitslag := StrToInt(tLine);

//V6.2.1 - 1
                    soek die pos van die bemanningslid met die bemanningID
                    pos := seekBemanningLid(bemanningID);

//V6.2.2a - 7
                    kontroleer of die bemanningslid 'n offisier is
                    if bSkik[pos] is TOffisier then
                        begin
                            skakel element om in 'n Offisier
                            f := bSkik[pos] as TOffisier;
                            slaag := false;
                            kontroleer of die offisier aan die vereistes voldoen het
                            korrek geneste if-stelling
                            if (f.kryRang = 1) AND (toetsUitslag >= 75) then
                                begin
                                    slaag := true;
                                end
                            else if (f.kryRang = 2) AND (toetsUitslag >= 80) then
                                begin
                                    slaag := true;
                                end
                            else if (f.kryRang = 3) AND (toetsUitslag >= 90) then
                                begin
```

```

        slaag := true;
    end
    else if (f.kryRang = 4) AND (toetsUitslag >= 95) then
    begin
        slaag := true;
    end;

    //V6.2.3 - 6
    6.2.3 - las die woord "Officer" en die naam aan
    output := output + 'Officer' + f.kryVolleNaam();

    //6.2.2a vervolg
    kontroleer of hy/sy geslaag het
    if slaag then
    begin
        bevorder die offisier
        f.bevorder();
        vervang objek in die skikking met die nuwe objek
        bSkik[pos] := f;

        6.2.3 - Las slaagteks vir offisier aan
        output := output + ' has passed and is promoted to '
                        + f.kryTitel() + ' Rank ' +
                        IntToStr(f.kryRang()) + #13#10;
    end
    else if f.kryRang() = 5 then
    begin
        6.2.3 - las kapteindeelname aan
        output := output + ' participated in the test ' +
                        #13#10;
    end
    else if slaag = false then
    begin
        6.2.3 - las druip vir 'n offisier aan
        output := output + ' has failed' + #13#10;
    end;
    end
    //6.2.2b - 6
    else
    begin
        kontroleer of die bemanningslid die minimum behaal het
        om te slaag en las slaag/druip aan
        if toetsUitslag >= 75 then
        begin
            inkrementeer getalBevorderdeBemanning
            Inc(TBemanningLid.getalBevorderdeBemanning);
            bevorder die bemanningslid deur nuwe offisier
            met bemanningslidbesonderhede te skep
            deur toepaslike datum te gebruik
            f := TOffisier.Create(bSkik[pos].kryVolleNaam() ,
bSkik[pos].kryBemanningID() , bSkik[pos].kryDepartement() , 1 ,
Now);

            voeg nuwe Offisier-objek by skikking by
            bSkik[pos] := f;

```

6.2.3 - Las slaagboodskap by nuwe offisier aan  
en sluit nuwe titel en rang in

```
output := output + f.kryVolleNaam() +  
            ' has passed and is promoted to ' +  
            f.kryTitel() + ' Rank ' +  
            IntToStr(f.kryRang()) + #13#10;  
end  
else  
begin  
    6.2.3 - Las druip vir 'n bemanningslid aan  
    output := output + bSkik[pos].kryVolleNaam + '  
            has failed' + #13#10;  
end;  
end;  
end;  
Result := output;  
end;
```

end;

end.

**VRAAG 5, 6.3, 6.4      uBemanningLidGK.pas**

```
program BemanningLidGK;
//V5.1 - 1
  Klasopskrif korrek
{$APPTYPE CONSOLE}
{$R *.res}

uses
  System.SysUtils,
  DateUtils,
  uBemanningLid in 'uBemanningLid.pas',
  uOffisier in 'uOffisier.pas',
  uBemanningLidBestuurder in 'uBemanningLidBestuurder.pas';

var

  cm : TBemanningLidBestuurder;

begin
  try
    { TODO -oUser -cConsole Main : Insert code here }

    //V5.2 - 2
    BemanningLidBestuurder-objek geskep
    op toepaslike plek
    cm := TBemanningLidBestuurder.create();

    //V5.3 - 2
    Roep naString en vertoon alle bemanningslede
    WriteLn(cm.naString());

    //V6.3 - 1
    Roep verwerkToetsUitslae in toepaslike klas
    WriteLn(cm.verwerkToetsUitslae);
    //V6.4 - 2
    Roep die statiese getalBevorderdeBemanning
    Roep die statiese getalBevorderdeOffisiere
    WriteLn('Number of promoted crew members ' +
      IntToStr(TBemanningLid.getalBevorderdeBemanning));
    WriteLn('Number of promoted officers ' +
      IntToStr(TOffisier.getalBevorderdeOffisiere));
    ReadLn;

  except
    on E: Exception do
      Writeln(E.ClassName, ': ', E.Message);
    end;
  end.
end.
```