	Vaknaam: Programmeren I (Java) - Tentamen	Richting: ICT
	Module: 2	Code: IC011
	Datum/Tijd: 17 mrt 2015 / 18.30 – 20:30	Docent: E. Liew
	Boeken en aantekeningen NIET toegestaan. Kladpapier is wel toegestaan.	
	De toets bestaat uit 4 opgaven. De puntenwaardering staat op de laatste bladzijde.	

Opgave 1

Gegeven de volgende Java code:

```
import javax.swing.*;
import java.awt.event.*;

// klasse Paneel
public class Paneel extends JPanel {
    private JButton knop;
    private JTextField tekstvak;

    public Paneel() {
        knop = new JButton("Klik");
        knop.addActionListener( new KnopHandler() );
        tekstvak = new JTextField(10);
        add(knop);
        add(tekstvak);
    }
    class KnopHandler implements ActionListener {
        public void actionPerformed(ActionEvent e) {
            tekstvak.setText("Je hebt geklikt!");
        }
    }
}
```

Bij de volgende beweringen moet je aangeven of ze WAAR of ONWAAR zijn:

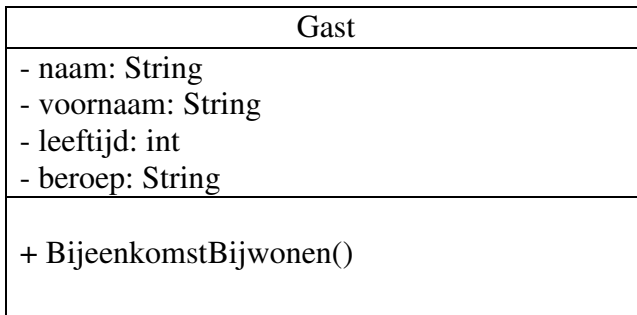
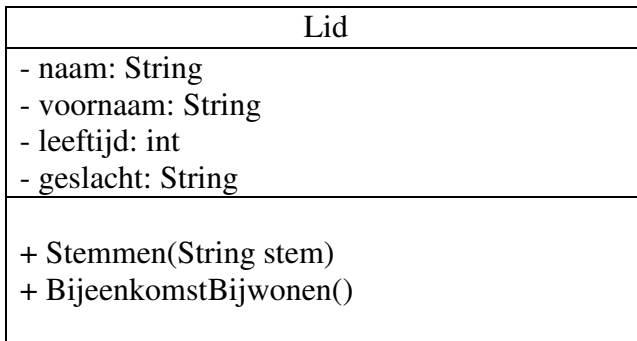
- De klasse *Paneel* maakt gebruik van Inheritance (overerving).
- In bovenstaande code komt er commentaar voor.
- KnopHandler* is een inwendige klasse van *Paneel*.
- Paneel* heeft 2 attributen.
- Paneel* heeft geen constructor methode.
- De klasse *Paneel* zorgt ervoor dat je *tekstvak* een waarde krijgt als je klikt op de *knop*.

Opgave 2

De politieke partij PP (Programmeer Partij) wil in verband met de komende verkiezingen haar ledenbestand automatiseren. Hiervoor is al het volgende uitgezocht en uitgewerkt:

- De partij heeft leden en sympathisanten.
- Van de leden wordt naam, voornaam, leeftijd en geslacht opgeslagen.
- Van sympathisanten wordt naam, voornaam, leeftijd en beroep opgeslagen.
- Leden kunnen stemmen bij bestuursvergaderingen.
- Sympathisanten mogen alleen partij bijeenkomsten aanwezig zijn.
- Leden mogen uiteraard ook partij bijeenkomsten bijwonen.

Van deze informatie zijn er 2 UML klassendiagrammen gemaakt.



Van deze twee klassen kan een superklasse *Persoon* definiëren. De klassen *Lid* en *Gast* moeten we dan wijzigen, zodat ze subklassen worden van *Persoon*. Deze principe heet overerving.

a) Maak de UML klassendiagrammen van de superklasse *Persoon* en haar subklassen *Lid* en *Gast*.

Opgave 3

Gegeven de volgende incomplete Java code:

```
public class Film {
    private String titel, genre;
    private int tijdsduur; // in minuten

    public Film(String titel, String genre, int tijdsduur) {
        this.titel = ...;
        this.genre = genre;
        - - - -
    }
    public String getTitel() {
        return titel;
    }
    ... int getTijdsduur() {
        return tijdsduur;
    }
    public void settTitel(... titel) {
        this.naam = naam;
    }
    public ... setGenre(String genre) {
        - - - -
    }
}
```

Er zijn 4 ontbrekende woorden en 2 ontbrekende regels.

... daar ontbreekt een woord.

- - - - daar ontbreekt een regel.

a) Maak bovenstaande code compleet.

b) Hoeveel attributen en methoden heeft de klasse Film?

```
public class Opdracht3 {
    public static void main(String[] args) {
        Film chappie = new Film("Chappie", "Science Fiction", 120);
        System.out.println(chappie.getTijdsduur());
    }
}
```

c) Wat is de uitvoer van bovenstaande regels code?

d) Breidt bovenstaande main methode uit, waarbij je een tweede object van de klasse Film aanmaakt. Vervolgens schrijf je de titel van het tweede object op je console scherm.

Het tweede object heeft de volgende waarden:

naam: Focus

genre: Action

tijdsduur: 133

Opgave 4

a) Schrijf de code die zorgt voor onderstaande uitvoer. Maak gebruik van de for statement.

```
2
33
444
5555
66666
777777
8888888
99999999
```

b) Schrijf de code die zorgt voor onderstaande uitvoer. Maak gebruik van de while statement.

```
100
90
80
70
60
50
40
30
20
10
0
```

c) Breid onderstaande code uit, zodat het de eerste 30 getallen van de rij van Fibonnaci op het console scherm schrijft.

De rij van Fibonnaci: 1,1,2,3,5,8,13,21 enzovoorts...

```
int a = 1;
int b = 1;
int som = 0;
int teller = 1;
```

Puntenwaardering

Opgave 1: 6 (1 + 1 + 1 + 1 + 1 + 1)

Opgave 2: 7

Opgave 3: 15 (8 + 2 + 1 + 4)

Opgave 4: 12 (4 + 3 + 5)

Totaal: 40/4 = 10