

Opgave 1.

a) Gegeven zijn de onderstaande classes

```
class A {
    int i;
```

```
    A (int i) {
        this.i = i;
    }
}
```

```
class B {
    A a;
```

```
    B (int x) {
        a = new A(x);
    }

    void put (int x) {
        a.i = x;
    }
}
```

en een programma dat de classes A en B gebruikt waarin de volgende declaraties/statements staan

```
A a = new A(3);
B b = new B(13);
```

```
void print (A a, B b) {
    out.printf("%d, %d\n", a.i, b.a.i);
}
```

```
void m1 (A p, int q) {
    p = new A(q);
}
```

```
void m2 (B p, int q) {
    p.a = new A(q);
}
```

```
void m3 (B p, int q) {
    p = new B(q);
}
```

```
void m4 (A a, int q) {
    a.i = q;
}
```

Wat is nu de uitvoer van een aanroep van de hierna gegeven methode doeleis() die in hetzelfde programma staat?

```
void doeleis () {
    print(a, b);
    m1(a, 5);
    print(a, b);
    m2(b, 8);
    print(a, b);
    m3(b, 2);
    print(a, b);
    m4(a, 1);
    print(a, b);
}
```

b) Gegeven is de onderstaande heading van een methode arcsin()

```
double arcsin (double x, int aantalTermen) // aantalTermen >= 1
```

Deze methode benadert m.b.v. "aantalTermen" termen onderstaande reeks

$$\arcsin(x) = x + \frac{1}{1 \cdot 2} (x^3/3) + \frac{1}{1 \cdot 3 \cdot 2 \cdot 4} (x^5/5) + \frac{1}{1 \cdot 3 \cdot 5 \cdot 2 \cdot 4 \cdot 6} (x^7/7) + \dots$$

$$= x + \frac{1}{1 \cdot 2} (x^3/3) + \frac{1}{3 \cdot 8} (x^5/5) + \frac{1}{15 \cdot 48} (x^7/7) + \dots$$

Hierbij is  $x^n$  de notatie voor " $x$  tot de macht  $n$ " en  $1 \cdot 3 \cdot 5$  de notatie voor  $1 \times 3 \times 5 (=15)$ . Het aantal uit te rekenen termen is gelijk aan de waarde van de parameter aantalTermen. Programmeer deze methode zonder gebruik te maken van methodes uit de class Math.

c) Declareer een variabele "matrix" die een tweedimensionaal array van integers met 57 rijen en 893 kolommen bevat. Doe dit gestructureerd, d.w.z. gebruik indien nodig constanten.

Schrijf een methode "stijgend" die voor een willekeurige matrix van int's kan controleren of de reeks van waarden van de rijssommen stijgend is. De rijssom van rij  $i$  is de som van alle waarden in rij  $i$ . De rijssommen zijn stijgend als de rijssom van rij  $i$  kleiner is dan de rijssom van rij  $i+1$  en deze weer kleiner is dan de rijssom van rij  $i+2$ , enz. t/m dat de rijssom van de voorlaatste rij kleiner is dan de rijssom van de laatste rij.

voorbeeld: De matrix 0 0 0 0 met rijssommen 0 is stijgend.

|   |   |    |    |    |
|---|---|----|----|----|
| 1 | 2 | 3  | 4  | 10 |
| 4 | 5 | 6  | 7  | 22 |
| 8 | 9 | 10 | 11 | 38 |

Gegeven is het programma in de onderstaande class:

```
import java.io.*;
class OpgaveId {
    PrintStream out;
    int f, g;
```

```
    OpgaveId() {
        out = new PrintStream(System.out); f = 3; g = 11;
    }
```

```
    int methode1 (int g) {
        int f = this.f + g;
```

```
        g /= 2;
        out.printf("%d\n%d\n", f, g);
        return this.f;
    }
```

```
    void methode2 (int f, int g) {
        this.f += f;
        g = this.g + 1;
        methode1(this.f);
        out.printf("%d\n%d\n", f, g);
    }
```

```
    void start() {
        f = methode1(5);
        methode2(g, f);
        out.printf("%d\n%d\n", f, g);
    }
```

```
    public static void main(String argv[]) {
        new OpgaveId().start();
    }
    Wat drukt dit programma af?
```

## Opgave 2.

- a) Gegeven zijn de onderstaande classes welke gebruikt kunnen worden voor het opslaan van gegevens van kunstwerken.

```
class Kunstwerk {
    String naamKunstenaar,
        soortKunstwerk;
    int inventarisnummer;
    double waarde;

    // De constructors en methodes doen er voor deze opgave niet toe.
}

class Museum {
    static final int MAX_AANTAL_KUNSTWERKEN = 12500;

    Kunstwerk[] kunstwerkArray;
    int aantalKunstwerken;

    ...
}
```

Schrijf een default constructor en een methode voegToe() voor de class Museum. De default constructor moet het Museum-object initialiseren op een leeg museum met 0 kunstwerken waaraan maximaal 12500 kunstwerken kunnen worden toegevoegd. De methode voegToe moet de mogelijkheid bieden om 1 Kunstwerk-object aan een Museum-object toe te voegen.

- b) Voorzie de class Museum van een methode

```
Museum schilderijen ()
```

welke in een nieuw Museum-object van alle in het museum aanwezige soorten kunstwerken de schilderijen retourneert. Een Kunstwerk-object bevat een schilderij als het veld soortKunstwerk gelijk is aan "schilderij".

Indien je een deelprobleem herkent, programmeer dit dan in een aparte methode in de juiste class.

- c) Gegeven is dat de class Museum een methode

```
Museum werk (String kunstenaar)
```

bevat. Deze methode retourneert in een nieuw Museum-object alle kunstwerken in het museum die van de meegegeven kunstenaar zijn.

Programmeer, gebruik makend van de methodes schilderijen() en werk() en zonder gebruik te maken van een herhalingsopdracht, een methode

```
boolean meesterschilder (String kunstenaar)
```

die retourneert of de meegegeven kunstenaar, op basis van de in het museum aanwezige schilderijen van deze kunstenaar, een meesterschilder is.

Voor deze opgave is gedefinieerd dat een kunstenaar een meesterschilder is indien het totale aantal schilderijen van deze kunstenaar in het museum meer dan 10 bedraagt.

- d) Schrijf voor de class Museum een methode

```
void verkoop (int inventarisnummer)
```

waarmee een kunstwerk kan worden verkocht, d.w.z. uit het museum verdwijnt.

## Opgave 3.

- a) Schrijf een recursieve functie

```
String keerOm (int n)
```

die voor een willekeurig niet-negatief geheel getal n, een String retourneert waar de cijfers van n in omgekeerde volgorde in staan.

voorbeelden:

```
keerOm(3).equals("3")
keerOm(123).equals("321")
keerOm(700).equals("007")
```

HINT: De concatenatie operator kan gebruikt worden voor de omzetting van een int in een String.

- b) Schrijf een recursieve methode

```
int codeer (String s)
```

die een tekst in een String s codeert in een getal. Gegeven is dat de String begint met een '(' en eindigt met een ')'. Verder is gegeven dat de String alleen hoofdletters en ronde haakjes bevat. Haakjes komen altijd in paren voor en kunnen genest zijn. Bijvoorbeeld "(AB(CD)E)".

Voor de codering van een hoofdletters wordt het volgnummer van die hoofdletter in het alfabet gebruikt. Deze code vermenigvuldigt met het volgnummer binnen het haakjespaar. De sommatie van de uitkomsten van deze vermenigvuldigingen levert de codering van het haakjespaar op.

HINT: 'C' - 'A' + 1 == 3

Voorbeeld: De codering van (FDE) is:

```
code (FDE) = 1*code F + 2*code D + 3*code E
           = 1*6 + 2*4 + 3*5
           = 29
```

Als een haakjespaar1 een ander haakjespaar2 bevat, wordt dit haakjespaar2 op de al beschreven manier gecodeerd en vermenigvuldigd met het volgnummer ervan binnen haakjespaar1. Hierbij telt haakjespaar2, net zoals een hoofdletter, voor 1 volgnummer.

Voorbeeld: De codering van (GED(FDE)A) is:

```
code (GED(FDE)A) = 1*7 + 2*5 + 3*4 + 4*code (FDE) + 5*1
                  = 1*7 + 2*5 + 3*4 + 4*29 + 5*1
                  = 150
```

En de code van (ZZ(GED(FDE)A)(FDE)) is:

```
code (ZZ(GED(FDE)A)(FDE)) = 1*26+2*26+3*code (GED(FDE)A)+4*code (FDE)
                           = 1*26+2*26+3*150+4*29
                           = 644
```

## Waardering

| Opgave | a | b | c | d | totaal |
|--------|---|---|---|---|--------|
| 1.     | 4 | 4 | 4 | 4 | 16     |
| 2.     | 2 | 4 | 3 | 3 | 12     |
| 3.     | 4 | 4 |   |   | 8      |

-- +

36

Het eindcijfer E volgt uit het puntentotaal T als volgt :  $E = T / 4 + 1$