

Opgave 1.

a) Gegeven zijn de onderstaande twee classes

```
class A {
    int i;

    A () {
        i = 0;
    }

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

    verdubbel () {
        i *= 2;
    }
}

class B {
    String s;

    B () {
        s = "";
    }

    B (String s) {
        this.s = s;
    }

    void verleng (String s) {
        this.s += s;
    }
}

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

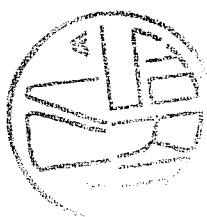
Output out = new Output();
A a = new A(5);
B b = new B("abcd");

void m1 (A par1, int par2) {
    par1.i += par2;
    par2 = 20;
}

void m2 (B par1, String par2) {
    par1.verleng(par2);
    par2 = "gh";
}

void print () {
    out.println(a.i);
    out.println(b.s);
}
```

Wat is nu de uitvoer van een aanroep van de hierna gegeven methode
doeleits() die in hetzelfde programma staat als de bovenstaande
declaraties?



```
void doeleits () {
    int i = 10;
    String s = "ef";
    print();
    m1(a, i);
    print();
    m2(b, s);
    print();
    m1(a, i);
    print();
    m2(b, s);
    print();
}
```

b) Schrijf voor positieve getallen de onderstaande methode

```
boolean perfectgetal (int i)    // i > 0
```

welke true retourneert als het meegegeven getal een perfect getal is.
Een perfect getal is een getal waarvoor geldt dat de som van alle
positieve gehele delers, die kleiner zijn dan het getal zelf, gelijk
is aan dit getal.

voorbeelden: 6 is een perfect getal want $1 + 2 + 3 = 6$
28 is een perfect getal want $1 + 2 + 4 + 7 + 14 = 28$

c) Schrijf de onderstaande methode merge()

```
int[][] merge (int[][] a, int[][] b)
/* gegeven is dat de matrices a en b hetzelfde aantal rijen
en kolommen hebben
*/
```

welke uit twee matrices een nieuwe matrix maakt door de overeenkomstige
elementen te mergen. Mergen betekent voor deze opgave dat het getal uit
a genomen wordt als de som van rij- en kolomindex even is en het getal
uit b genomen wordt als dit niet het geval is.

Voorbeeld:
bif a = 1 2 3 en b = -1 -2 -3 levert merge(a, b) dan 1 -2 3 op

4	5	6	-4	-5	-6	-4	5	-6
7	8	9	-7	-8	-9	7	-8	9

Z.O.Z.