

P1W6 Opdracht Dobbelstenen

Stap 1:

Download het bestand **Dobbelstenen.zip** en pak het uit in een afzonderlijke directory.

Dit is een compleet afgewerkt programma. Compileer het programma en voer het uit.

Bekijk de drie aanwezige klassen. In de klasse **Dobbelstenen** wordt er gebruik gemaakt van objecten van de klasse **Dobbelsteen** en in de klasse **TestDobbelstenen** wordt gebruik gemaakt van één object van de klasse **Dobbelstenen**.

Stap 2:

Op dit ogenblik bevat de klasse **Dobbelstenen** alleen een *default constructor* en de methoden **werpDobbelsteen**, **werpTweeDobbelstenen** en **werpDrieDobbelstenen**.

Kijk dit na een bekijk ook eens wat er allemaal in de **main**-methode van de klasse **TestDobbelstenen** staat. Dit kan beter!

De bedoeling van deze opdracht is om te komen tot de volgende inhoud van de klasse **TestDobbelstenen**:

```
public class TestDobbelstenen {
    private static final int AANTAL_WORPEN = 100;

    public static void main(String[] args) {
        Dobbelstenen stenen = new Dobbelstenen(AANTAL_WORPEN);

        System.out.println(AANTAL_WORPEN + " worpen met 1 dobbelsteen:\n");
        stenen.werpDobbelsteen();
        System.out.println(stenen);

        System.out.println(AANTAL_WORPEN + " worpen met 2 dobbelstenen:\n");
        stenen.werpTweeDobbelstenen();
        System.out.println(stenen);

        System.out.println(AANTAL_WORPEN + " worpen met 3 dobbelstenen:\n");
        stenen.werpDrieDobbelstenen();
        System.out.println(stenen);
    }
}
```

Hiervoor dienen er in de klasse **Dobbelstenen** een aantal aanpassingen te gebeuren:

- Voeg een constante **AANTAL_WORPEN** toe maar geef ze nog geen waarde
- Voeg een tabel (reeks) voor gehele getallen toe met de naam **worpen** (doe alleen de declaratie)
- Pas nu de *constructor* aan zodat hij een parameter van het type **int** meekrijgt met de naam **aantal**. De constante **AANTAL_WORPEN** moet de waarde van deze parameter krijgen. Verder moet de tabel **worpen** geïnitieerd worden zodat hij **AANTAL_WORPEN** kan bevatten.

- De drie bestaande methoden **werpDobbelsteen**, **werpTweeDobbelstenen** en **werpDriedobbelstenen** moeten nu als returnwaarde **void** krijgen. In plaats van een getal terug te geven moeten ze **AANTAL_WORPEN** uitvoeren en elk van de worpen in volgorde in de tabel **worpen** plaatsen.
- De laatste methode die we dienen toe te voegen is een **toString** methode. Deze methode zullen we gebruiken om telkens de 100 verschillende worpen te tonen. Omdat je met de tabel **worpen** werkt kan je deze **toString** methode gebruiken bij het werpen met 1, met 2 en met 3 dobbelstenen. Een duidelijk voorbeeld van hergebruik van code en één van de vele voordelen van het gebruik van klassen. Bij de uitwerking van de **toString** methode is het nuttig om eens te kijken naar een klassenmethode van de klasse **String**, met name de **format**-methode (dit om de gebruikte **printf**-methode bij de string *concatenation* te vervangen). Verder moet je ervoor zorgen dat er na 10 getallen telkens een *newline* wordt toegevoegd.

De afdruk zou er na uitvoering van het programma nu als volgt moeten uitzien (uiteraard met andere getallen):

100 worpen met 1 dobbelsteen:

```
5 4 6 2 6 6 2 4 2 1
1 1 6 5 6 5 6 5 5 6
5 6 4 5 1 4 1 6 1 1
2 6 5 1 3 5 3 3 1 6
2 4 2 2 1 3 4 4 6 4
1 5 3 4 2 6 5 3 1 4
6 6 2 6 2 3 2 6 2 1
3 5 4 1 1 6 5 6 4 6
4 1 3 4 5 5 5 4 1 4
1 3 6 3 4 3 1 1 4 2
```

100 worpen met 2 dobbelstenen:

```
10 8 9 5 4 9 12 11 7 5
5 6 9 7 5 7 5 8 7 7
7 4 8 7 6 6 3 6 4 3
10 6 7 6 6 10 5 10 6 8
8 4 5 4 7 6 5 6 9 11
4 3 11 4 3 9 4 6 5 8
6 5 10 8 7 10 10 10 5 8
2 8 5 5 5 5 11 5 9 4
2 6 3 9 5 10 5 11 6 8
5 11 6 5 7 9 5 9 11 5
```

100 worpen met 3 dobbelstenen:

```
13 10 13 12 5 13 11 11 10 14
10 6 8 9 13 7 12 12 5 16
12 6 11 15 15 9 13 9 10 13
9 18 9 7 11 8 8 8 11 10
7 14 9 16 13 13 7 5 9 8
10 9 11 11 12 14 12 13 9 10
10 11 11 12 15 12 10 9 12 8
10 13 10 10 14 16 9 12 8 11
8 6 10 8 15 8 7 16 13 10
13 10 13 6 5 9 13 6 14 12
*/
```

Stap 3:

Voeg nu een **private** methode met de naam **berekenGemiddelde** toe. Deze methode is dus een methode die alleen intern in de klasse **Dobbelstenen** gebruikt wordt.

De methode geeft een **double** terug met de gemiddelde waarde van de 100 dobbelsteenworpen.

Maak in de **toString** methode gebruik van deze methode om ook telkens de regel met de gemiddelde waarde te kunnen tonen.

Nu zou de afdruk er terug uit moeten zien zoals die van het oorspronkelijke programma (uiteraard met andere cijfersequenties).