

# Maximale driehoekssom

## Programmeeropdracht 1 Algoritmen en Complexiteit 2016

### 1 Probleembeschrijving

We bekijken piramides van getallen waarin op de eerste regel één getal staat, op de tweede regel twee getallen, op de derde regel drie getallen, et cetera. Ontwerp een programma dat een pad vindt in deze driehoek, van de bovenste regel tot de onderste regel, zodanig dat de som van de getallen in dit pad maximaal is. Het volgende getal in een pad ligt ofwel direct linksonder het huidige gekozen getal ofwel direct rechtsonder het huidige gekozen getal.

Bekijk bijvoorbeeld de volgende driehoek:

```
  3
 7 5
1 4 5
9 5 8 2
```

Het vetgedrukte pad is in deze driehoek het pad met de maximale som.

### 2 Invoer/uitvoer

De getallen in de piramide zijn strikt tussen 0 en 100 en worden indien nodig van extra nullen voorzien aan het begin. Als uitvoer wordt de som van het maximale pad verwacht.

#### 2.1 Voorbeeldinvoer

```
03
07 05
01 04 05
09 05 08 02
```

## 2.2 Voorbeelduitvoer

22

## 3 Inleveren

**Let op:** Deze programmeeropdracht wordt voor een groot deel automatisch nagekeken. Zorg dus dat je je precies aan onderstaande richtlijnen houdt, anders kan je code niet worden nagekeken.

### Controleer of je code goed werkt voor het inleveren

Door `check.sh` uit te voeren in dezelfde map als je python- en input/output-files kun je controleren of je code de juiste uitvoer geeft op de kleine input (`small.input`) en het juiste inputformaat heeft.

### Nakijkeisen

- De code dient geschreven te zijn in Python 3.
- De code leest de input van *stdin* en schrijft naar *stdout*. Om eenvoudig invoer te testen kun je gebruikmaken van input redirection:  
`python ex1.py < small.input`

*Hint: de volgende voorbeeldcode leest op de juiste manier van stdin en schrijft hetzelfde naar stdout.*

```
import fileinput
for line in fileinput.input():
    # line bevat al een newline van zichzelf
    print(line, end="")
```

- Het codebestand zelf moet de naam `ex1.py` hebben.
- Lever **alleen** het python bestand met je code in en geef het de naam `ex1.py`. Lever dus geen zip/tar in of iets dergelijks.

### Hoe wordt de code nagekeken?

Er zijn twee inputfiles aan de hand waarvan je code wordt nagekeken:

- Bijgeleverd is `small.input` met de bijbehorende output `small.output`. Als je code dezelfde output geeft met `small.input` als input, wordt in principe het cijfer 6 toegekend.

*NB: Om deze score te behalen is het wel noodzakelijk dat je daadwerkelijk redelijkerwijs het algoritme hebt geïmplementeerd. Simpelweg altijd dezelfde output retourneren wordt niet goedgekeurd.*

- Daarnaast is ook voor elke student apart (in `large.input.zip`) een inputfile `large_{studentnummer}.input` bijgeleverd (zonder de bijbehorende output). Als je code ook de juiste output voor deze input geeft, wordt in principe het cijfer 10 toegekend.

De code wordt hiernaast gecontroleerd op plagiaat van anderen en code op het internet. Hierop is de algemene plagiaat- en frauderegeling van de UvA van toepassing.

### **Geen inputfile ontvangen / overige vragen**

Mocht je studentnummer niet tussen de inputfiles staan of mocht je overige vragen/problemen hebben over het inputformaat, stuur dan een mail naar [R.B.Kluiving@uva.nl](mailto:R.B.Kluiving@uva.nl).