

Toets Parel 001 — Algoritmiek

13 september 2013

Dit tentamen bestaat uit 7 vragen. Het cijfer voor deze toets is het aantal punten gedeeld door 10.

- 1 (10 punten) Stel dat `p` de volgende lijst bevat: `[[5, "vijf"], [6, "zes"], [10, "zeven"]]`.
- Schrijf een conditie in Python-syntax die test of het tweede element van het paar met index `i` gelijk is aan het tweede element van het paar met index `j`. (Bijvoorbeeld, als `i` op 0 staat en `j` op 2 dan vergelijk je "vijf" met "zeven".)
 - Schrijf een enkele toewijzing in Python-syntax die ervoor zorgt dat het woord "zeven" vervangen wordt door "tien".

Analyseer de volgende Python-functie:

```
1 def split(data):
2     even = []
3     odd = []
4     i = 0
5     while i < len(data):
6         if data[i] % 2 == 0: als even
7             even.append(data[i])
8         else: als oneven.
9             odd.append(data[i])
10            i = i+1
11    return [odd, even]
```

(De operator `%` in regel 6 staat voor *modulo*: de conditie `data[i] % 2 == 0` test dus of `data[i]` een even getal is.)

- 2 (15 punten) Geef aan wat er gebeurt bij de aanroep `split([3, 6, 2, 10, 1])`, door een tabel te tekenen waarin telkens de waarde van `i`, `odd` en `even` direct vóór het uitvoeren van regel 10 staat. Wat is het eindresultaat van de aanroep?
- 3 (15 punten) Hoeveel stappen heeft `split` nodig als `data` een lijst van n elementen is: ongeveer ("orde-grootte") n , ongeveer $n \log_2 n$ of ongeveer n^2 ? Beredeneer je antwoord.
- 4 (10 punten) Wat zou er gebeuren als regel 10 uit `split` weggehaald zou worden?
- 5 (20 punten) Beschouw de lijst `[10, 8, 4, 11, 3, 4]`.
- Laat zien hoe bubblesort deze lijst sorteert, door middel van een schema waarin de lijst na elke verandering is opgenomen.
 - Laat zien hoe mergesort deze lijst sorteert, door middel van een schema waaruit duidelijk wordt hoe de lijst gesplitst en weer aan elkaar geritst wordt.
- 6 (10 punten) Stel dat je een lijst woorden `woord1` hebt die helemaal gesorteerd is. Nu maak je een nieuwe lijst door het eerste element van `woord1` te vervangen door het woord "zebra".
- Welk algoritme sorteert `woord1` sneller: bubblesort of mergesort? Verklaar je antwoord.
- 7 (20 punten) Je hebt een boekenplank vol boeken, geordend van klein naar groot. Op een dag word je wakker en beseft je dat je ze nu eens van groot naar klein gesorteerd wilt hebben.
- Beschrijf een algoritme, met eenduidige (eventueel genummerde) instructies, dat je zo snel mogelijk naar de gewenste situatie voert. (Je mag ervan uitgaan dat je genoeg ruimte hebt om boeken tijdelijk neer te zetten.)
 - Hoeveel stappen heeft je algoritme nodig, uitgedrukt in termen van het aantal boeken op de plank?

