

Sommersemester 2011

## Übungen zu Informatik II - Blatt 10

Abgabe am 30.06

### Organisatorisches

- Die theoretischen Aufgaben müssen Sie donnerstags in der Vorlesung abgeben.
- Die Programmieraufgaben müssen Sie donnerstags bis spätestens 13:15 Uhr an Ihren Tutor per Email ([christian.schnarr@tu-clausthal.de](mailto:christian.schnarr@tu-clausthal.de)) schicken.
- Die Programmieraufgaben müssen von Ihnen in der Übung vorgeführt und erklärt werden.

### Aufgabe 1 (Greedy, ASP, 2 Punkte)

Wenn man den Greedy-Algorithmus zur Lösung des Activity-Selection-Problems ein klein wenig modifiziert, nämlich die Liste der Activities zu Beginn nach der Startzeit sortiert (anstatt nach der Endzeit), dann würde der Algorithmus nicht notwendigerweise eine optimale Lösung erzeugen. Warum?

### Aufgabe 2 (Greedy-Algorithmen, 5 Punkte)

Gegeben sind zwei Zahlenfolgen  $\mathcal{A} = (a_1, \dots, a_n)$  und  $\mathcal{B} = (b_1, \dots, b_n)$ , welche je  $n$  positive Integer-Zahlen enthalten. Sei weiterhin  $\sigma : [1, n] \rightarrow [1, n]$  eine Permutationsfunktion der Indices von  $\mathcal{B}$ . Geben Sie einen Greedy-Algorithmus an, welcher

$$\prod_{i=1}^n a_i^{b_{\sigma(i)}} \quad (1)$$

maximiert, d.h., das optimale  $\sigma$  bzgl. Formel 1 konstruiert.

Zeigen Sie, daß Ihr Algorithmus die *Greedy-Choice*-Eigenschaft erfüllt.

### Aufgabe 3 (Autofahren mit Greedy-Algorithmen, 8 Punkte)

Mit Ihrem neuen Erdgas-Auto planen Sie einen Ausflug über die Alpen zum Schiefen Turm von Pisa. Einerseits wollen Sie auf der Fahrt so selten wie möglich zum Tanken anhalten. Andererseits müssen Sie beachten, dass bisher nur ausgewählte Tankstellen Erdgas anbieten. Mit einem vollem Tank können Sie maximal  $D$  Kilometer fahren. Sie starten mit vollem Tank in Clausthal-Zellerfeld ( $E_0$ ) und fahren bis nach Pisa ( $E_n$ ). Unterwegs kommen Sie der Reihe nach an den Erdgas-Tankstellen  $E_1, E_2, \dots, E_{n-1}$  vorbei. Der Abstand zwischen  $E_{i-1}$  und  $E_i$  beträgt  $d_i$  Kilometer (mit  $i = 1, \dots, n$ ), wobei jeweils  $d_i \leq D$  gilt.

a) Geben Sie eine Greedy-Strategie für dieses Problem an.

- b) Implementieren Sie Ihre Strategie in Python, so dass Ihr Programm die minimale Anzahl an Tank-Stops ermittelt und eine Liste der anzufahrenden Tankstellen zurückgibt (der Index der Tankstelle im Array ist hier ausreichend).

```
#####  
# Implementieren Sie ihren Ansatz so, dass beim Aufruf:      #  
#                                                           #  
# array_Tankstellen = IHRE_STRATEGIE ( D,d )                #  
#                                                           #  
# ein Array mit den anzufahrenden Tankstellen ausgegeben wird.##  
#####  
#  
# Beispiel:  
#  
# Reichweite mit einer Tankfuellung  
D = 400  
# Bsp. Array fuer Abstaende der Tankstellen, beachten Sie, dass  
# fuer jeden Wert im Array d[i] < D gelten muss  
d = [100,20,304,50,43,34,2,34,124,45,232,45,90]  
meine_Tankstops = berechne ( D,d )  
print meine_Tankstops  
# Lsg. --> meine_Tankstops = [1,4,9]
```

- c) Warum hat Ihre Greedy-Strategie die Greedy-Choice-Eigenschaft, d.h., warum liefert sie eine optimale Lösung?