



Záverečný test

Zadanie



Ústav informatiky
Prírodovedecká fakulta
UPJŠ v Košiciach

Doplňujúce zdrojové kódy sú na stránke predmetu PAZ1b. Funkčnosť každého riešenia musí byť preukázaná spustením na testovacích vstupoch - nespustiteľné riešenia neumožňujú získanie príslušných bodov.

Duna

Planéta Arrakis, tiež zvaná Duna, je drsná púštna planéta ale hlavne je jediným zdrojom korenia v galaxii. Bez korenia navigátori nemôžu cestovať medzihviezdny priestorom nadsvetelnou rýchlosťou, nemôže sa vyrábať mnoho liekov ktoré predlžujú život. Imperátor aj veľkorody vedia, že „Who controls the spice controls the universe“. Ťažba korenia a jeho rafinácii sa venuje skoro každý na planéte. Veľkorod Harkonenovcov utláčateľov pre ktorých je to najväčším zdrojom bohatstva. Fremeni, jediné pôvodné obyvateľstvo planéty a jediný kto dokáže prežiť v nehostinnej púšti. Pašerácke skupiny ktoré za krátku dobu môžu zbohatnúť na niekoľko životov. Okrem nich planétu brázdia obrovské červy, ktoré dosahujú až dĺžku vyše jedného kilometra a nie je známy spôsob ako ich zastaviť. Ale aj červy sa vyhýbajú púštnym búrkam v ktorých sa piesok pohybuje takou rýchlosťou, že reže oceľ.

Zber korenia (7+3+4+5 bodov, backtracking)

Zberače korenia sú naplnené korením zmiešaným s pieskom. Dnes sa stíha prepraviť korenie do vzdialenej rafinérie až dvanásťkrát (vo všeobecnosti k -krát). Prepravná helikoptéra (ornitoptera) vždy zoberie celý náklad zberača, pričom môže zobrať náklad z viacerých zberačov ale len náklad do maximálnej prepravnej hmotnosti. O každom zberači vieme akú váhu má náklad zberača a koľko z nákladu je korenie. Do rafinérie, kde sa oddelí korenie od piesku, chceme previesť čo najviac korenia. Navrhnete algoritmus, ktorý vypočíta, aké najväčšie množstvo korenia je možné prepraviť.



Úlohy: Implementujte program ktorý, načíta nosnosť helikoptéry h , počet prevozov, ktoré budeme realizovať k a informácie o hmotnosti nákladov jednotlivých zberačov celková váha a objem korenia (formát vstupu si zvolte podľa vlastného uváženia).

(7b) Program vypíše, aké najväčšie množstvo korenia je možné dopraviť do rafinérie na k prepráv.

(+3b) Za výpis, ktorý určí pri ktorej preprave sa vyprázdnia ktoré zberače.

(+4b) Upravte program tak, aby zistil na aký počet prepráv helikoptérou je možné vyprázdniť všetky zberače.

Pozn.: Buď riešte túto úlohu samostatne od predchádzajúcich dvoch úloh alebo ju implementujte tak, aby bolo možné stále ich spúšťať.

(+5b) Podľa efektívnosti. Algoritmus je tým efektívnejší, čím viac eliminujete výpočty, ktoré nevedú k prípustnému riešeniu.

Rada: Uvedomte si, že každý spôsob zberu korenia je určený poradím vyprázdnenia zberačov.

Harkonenské légie pochodujú púšťou (6+3+8 bodov, grafové algoritmy)

Harkonenské légie sa vybrali obsadiť ďalšiu rafinériu. Púšť je ale príliš nehostinná a preto putujú medzi jednotlivými úkrytmi iba v noci. Navyše nie každý úkryt má aj zásobu vody. Légia nemôže spať dva dni za sebou v úkryte bez zásoby vody. Preto je putovanie náročné na prípravu správnej cesty.



Úlohy:

(6b) Implementujte program, ktorý načíta graf medzi vrcholmi ktorého sa dokáže légia presunúť za jednu noc. Reprezentáciu grafu a formu vstupu si zvolíte podľa vlastnej preferencie. Program načíta vrchol v ktorej sa légia nachádza, načíta vrchol kde je rafinéria, do ktorej sa má légia dostať a načíta, ktoré úkryty majú zásobu vody. Algoritmus vypočíta najkratšiu cestu (počet nocí) za ktorú sa légia dostane zo štartovacieho do cieľového vrcholu.

(+3b) Algoritmus vypíše postupnosť vrcholov na najkratšej ceste.

(+8b) Upravte program tak, aby našiel najkratšiu cestu za predpokladu, že légia môže byť bez vody až 2 dni (prespania).

Evakuácia (3+5 bodov, greedy + ???)

Fremeni sa rozhodli evakuovať svoj úkryt lebo sa doň blíži Harkonenská légia. Potrebujú zachrániť hlavne čo najviac veľkého vybavenia. Každý môže zobrať nanajvýš jeden kus vybavenia a jedno vybavenie nemôže niesť viac ľudí. O každom človeku v úkryte vedia, aký najťažší kus vybavenia dokáže uniesť a majú aj zoznam vybavení a ich hmotnosti.



Úlohy: (3b) Navrhňte algoritmus, ktorý načíta nosnosti jednotlivých ľudí a váhy zariadení. Algoritmus vypočíta najviac koľko kg zariadení je možné preniesť. Očakávané riešenie je v čase $O((n.m))$, kde n je počet ľudí a m je počet zariadení. Nezabudnite mať pripravené argumenty, že vaše riešenie vypočíta maximálnu prenesenú hmotnosť.

Nosnosti
40, 10, 20

Zariadenia
50, 5, 35, 10, 10, 15

Maximálna prenesená hmotnosť 60kg (35+10+15).

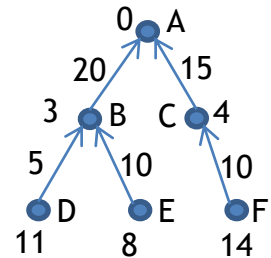
(+5b) Za riešenie s časovou zložitnosťou lepšou ako $O(n.m)$, pričom môžete predpokladať, že m a n sú podobne veľké ($n=m$). Nezabudnite mať pripravené argumenty o časovej zložitosti.

Zberače rosy I (7 bodov stromy/orientované grafy)

V síči (úkryte fremenov) majú high-tech zberače rosy, ktoré ich zásobujú vodou. Podľa toho ako prúdi ráno vzduch nazbierajú rôzne množstvo vody. Aby nemuseli každý deň chodiť po vodu, tak sú zberače pospájané rúrkami, ktoré privedú vodu až na jedno zberné miesto. Všimli si, že niektoré dni dotečie na zberné miesto menej vody ako zberače signalizujú, že sa jej podarilo zozbierať, teda systém rúrok má niekde nedostatočný prietok. O každej rúrke, ktorá spája nejakú dvojicu zberačov vedia koľko vody ňou môže pretečť a o každom zberači rosy vedia koľko vody vyprodukuje.

Úloha: Napíšte program, ktorý načíta produkcie zberačov rosy, potrubia medzi zberačmi rosy a prietok každého potrubia. Program vypíše koľko vody dotečie na zberné miesto. Formát vstupu si zvolte podľa vlastného uváženia. Pozn. graf (sieť) potrubí tvorí strom. Pre jednoduchosť predpokladajte, že zberné miesto je zberač s nulovou produkciou.

Na príklade vpravo, do zberného miesta dotečie 30l vody. Lebo z D do B pretečie 5, z E do B 8, z B do A 16, z F do C 10, z vrcholu C do A 14.



Zberače rosy II (4 až 12 bodov, dynamické programovanie)

Čím je zberač rosy väčší tým je efektívnejší. Ale aby zberač rosy nespapal púštny červ musí byť postavený na skalách a nie na piesku. Fremeni si preto urobili mapu skál, kde by chceli postaviť čo najväčší zberač rosy. Napr. mapa vyzerá nasledovne (ak potrebujete, tak súbor môže obsahovať na začiatku aj rozmery mapy):

```
00000
01111
01110
11110
```

Vzhľadom k prúdeniu vetru a technológií má zberač špecifický tvar a orientáciu (ako trojuholník s podstavou dole). Tri zberače najmenších veľkostí vyzerajú nasledovne (X vyjadruje zberač).

```
1      2          3
          X
        XXX
X     XXX     XXXXX
```

Možné riešenia vyznačené v mape

```
00000      00000      00000
01X11      01111      01111
0XXX0      0X110      01X10
11110      XXX10      1XXX0
```



Úloha: Navrhňte program, ktorý načíta mapu a vypočíta veľkosť najväčšieho zberača, ktorý môže byť postavený.

(10b) za riešenie v čase $O(n)$, ku každému políčku pristúpime len pevný počet krát (dynamické programovanie). (4b) za iné riešenie. (+2b) za zobrazenie najväčšieho zberača na mape.

Harkonenské dane (4 +4 bodov, spájané zoznamy)

O Fremenoch všetci predpokladali, že nemajú veľké bohatstvo a nie sú veľmi schopní. Ale aj tak sa ich snažili Harkonenovci zdanit'. Aby mohli Fremeni prejsť do mesta bez platenia dane, tak išli za sebou tak aby nikdy traja (vo všeobecnosti k -ti) za sebou nemali priveľa peňazí.

Úloha: (4b) Uvažujme triedu SpajanyZoznam z prednášky o spájaných zoznamoch. Do kódu z prednášky pridajte metódu, ktorá vráti maximum k čísel idúcich za sebou v spájanom zozname. Očakávané riešenie je v čase $O(n)$ a nie je možné si zoznam prekopírovať do inej štruktúry. (+4 body) za efektívnosť. 2b za k prístupov ku každému prvku spájaného zoznamu. 4b za konštantný počet prístupov ku každému prvku spájaného zoznamu.

Pr. $k = 3$, spájaný zoznam [5,5,5,5,5,1,3,2,5,8,0,8,3,4] metóda vráti 16,
 $k = 2$, spájaný zoznam [5,5,5,5,5,1,3,2,5,8,0,8,3,4] metóda vráti 11,
 $k = 5$, spájaný zoznam [5,5,5,5,5,1,3,2,5,8,0,8,3,4] metóda vráti 25.