

**Korešpondenčný seminár z programovania
XXII. ročník, 2004/05**
Katedra základov a vyučovania informatiky FMFI UK,
Mlynská Dolina, 842 48 Bratislava

*KSP finančne podporujú: aSc – Applied Software Consultants spol. s r.o.
MICROSTEP-MIS spol. s r.o.*

Príklady 1. kola zimnej časti, kat. Z

Táto veta pravdivá je: KSP skrýva rôzne tajy. Dávno každý vie, že riešiť ho je kúzelné... A tak deti neváhajte, príklady nám posielajte. Termín je **5. január 2005**, keď príde, do obálky ich vložte hneď. A nezabudnite na známky za 15 korún, nech dostanete od nás odozvu skorú.

„James, povedz muchám na stene, že Lord Norton im vznešene odkazuje HEŠ!“

KSPáci

1. Zruční artisti II

15 bodov

Kaledonskí cirkusanti potrebujú znovu Vašu pomoc! Aj keď ste im pomohli minule, tak to nestačilo. Vymysleli novú zostavu. Tentoraz chcú postaviť „pyramídu“. Ich pyramída má vyzeráť tak, že dvaja artisti držia na pleciach tretieho. Výsledok bude vyzeráť tak, že na spodku pyramídy je K artistov. Na ich pleciach je $K/2$ artistov, pričom dvaja držia práve jedného. Na „treťom“ poschodí je $K/4$ artistov. Tak to pokračuje, až je navrchu jediný. Aj tu platí, že artista, ktorý je navrchu, musí byť ľahší ako každý z dvoch kolegov ktorí ho držia.

Úloha: Na vstupe je zadané prirodzené číslo N , pričom platí, že $N + 1$ je mocninou 2. Ďalej nasleduje N čísel, ktoré vyjadrujú hmotnosť každého artistu. To je zároveň aj najväčšia hmotnosť artistu, ktorý mu môže stáť na pleciach. (Stačí ak artista unesie toho priamo nad sebou, na hmotnosti ostatných artistov nad ním nezáleží.)

Vašou úlohou je vypísať hmotnosti artistov v takom poradí, v akom majú vytvoriť „pyramídu“, ktorá nespadne. Výslednú pyramídu vypíšete postupne po úrovniach, začínajúc najvyššou. Stačí nájsť ľubovoľné jedno prípustné riešenie.

Príklad:

Vstup:

$N=7$

váhy: 62 70 55 68 66 74 47

Výstup:

47 55 68 66 62 70 74

Poznámka. Pyramída vyzerá nasledovne: Artista s váhou 47 stojí na pleciach artistov s váhami 55 a 68. Artista s váhou 55 stojí na pleciach artistov s váhami 66 a 62. Artista s váhou 68 stojí na pleciach posledných dvoch artistov.

2. Zábava na kolieskach II.

15 bodov

Hoc ste minule Jančimu pomohli viac než dosť, (čím ste si ho veľmi zaviazali) objavil sa problém druhý... Nie preto, že by sa korčuľovať nevedel, ale práve preto, že sa korčuľovať vie... a hlavne - korčuľuje! Teraz v zime! „Ako blázon, premáva sa na korčuliach sadom. Krížom-krážom, cez kaluže vody špinavej, prebýja sa hĺbou lístia cez alej.“ Ale späť k problému. Tým teraz nie je ani tak Jančiho chorá myseľ či zlomené srdce, ako priveľké množstvo rôznych nečistôt po ceste. Špina nepôsobí na korčule príliš blahodárne, na rýchlosť pohybu taktiež. Nezmyselnosť jeho konania mu určite nevysvetlíte, no môžete (musíte!) mu poradiť tú „vhodnejšiu cestu“. Spomedzi najrýchlejších možných takú, kde prejde cez najmenší objem nečistôt.

Úloha: Máme zadanú mapu rozmerov $N \times M$ políček, pričom každé políčko je buď nepriechodné “#”, alebo obsahuje jeden z K typov priechodných povrchov. Pri povrchu sa uvádza objem nečistôt (H - celé číslo od 0 do 10) daného povrchu - spôsobujúce pri prechode cez dané políčko zdržanie o H minút. Normálna rýchlosť je políčko za minútu, ak má políčko $H = 3$, potom políčko prejde za 4 minúty. Ďalej sú v celej mape 2 znaky, “J” ako aktuálna pozícia Jančiho, “D” ako domov. Vašou úlohou je opäť vypísať čo možno najrýchlejšiu cestu, a ak je ich viac, tak tú s najmenším objemom nečistôt.

Ak je táto úloha na vás príťažká, pár bodov dostanete aj vtedy, ak len nájdete ľubovoľnú najrýchlejšiu cestu – teda akoby ho niektoré povrchy spomaľovali, ale nečistoty nezbieral.

Príklad:

Vstup:

30x15

5

. 0

* 1

+ 2

@ 3

% 4

```
#####
#+. . . . . @ % % # % % @ % . . . . . + + #
# + # . . . . . @ % % # % % @ % . . . . . # + #
##### * @ % % # % % @ % #####
# . . ##### * @ % % # % % @ % ##### . . #
# . . ##### * @ % % # % % @ % ##### . . #
# . . * ##### * ##### * ##### + + . . #
# . . * ##### * ##### * ##### D + * . . #
# . J * ##### * ##### * ##### + + . . #
# . . ##### * @ % % # % % @ % ##### . . #
# . . ##### * @ % % # % % @ % ##### . . #
##### * @ % % # % % @ % ##### + + #
# + # . . . . . @ % % # % % @ % . . . . . # + + #
# + + . . . . . @ % % # % % @ % . . . . . + + + #
#####
```

Výstup:

```
#####
# + + - - - - - \ . @ % % # % % @ % . - - - - - \ + + #
# + / # . . . . . \ . @ % % # % % @ % . / . . . . . # \ + + #
# / ##### * \ * @ % % # % % @ % / ##### * # | #
# | . ##### * | @ % % # % % @ / ##### * # | #
# | . ##### * \ @ % % # % % @ | ##### * # / #
# . \ * ##### * - - - - - / * ##### * # + + / . #
# . | * ##### * ##### * ##### * D - - . . #
# . J * ##### * ##### * ##### * # + + . . #
# . . ##### * @ % % # % % @ % ##### * # . . #
# . . ##### * @ % % # % % @ % ##### * # . . #
##### * @ % % # % % @ % ##### * # + + #
# + + # . . . . . @ % % # % % @ % . . . . . # + + + #
# + + . . . . . @ % % # % % @ % . . . . . + + + #
#####
```

3. Zakódované dáta

15 bodov

Došiel raz jedného krásneho slnečného popoludnia Janík zo školy domov a pri pohľade na krásne zelené lesy navôkol, ktorých sa zatiaľ žiadna veterná smršť ani len nedotkla, rozhodol sa, že si on vyjsť von do prírody musí, preč od počítača. Len keby nemusel úlohy a hlavne štvorstranový referát do školy písať. I zavolať on kamarátovi Miškovi a natešený, že mu ten poslať svoj referát emailom prisľúbil, utekal von.

Ked sa večer Janík vrátil, zobral sa, že trochu referát od Miška pozmení, aby príliš okaté nebolo, že ho odpísal. Ale beda, súbor od Miška vôbec nevyzeral ako normálny text a Janík preľaknutý, že si nebudaj Miško z neho srandu robí, alebo mu referát zašifrovaný poslal, rýchlo zdvihol telefón. Miško mu však len vysvetlil, že on len kvoli veľkosti a prenosovej rýchlosti súbor svojím vlastným programom skomprimoval, a že Janík si s dekomprimáciou určite ľahko poradí.

I zaprisahal sa Janík, že on už nikdy v živote nebude podvádzať a bude si pekne poctivo úlohy a referáty sám písať. Len na tento jeden už ani za mak času nemá, a tak by rád vašu pomoc privítal.

Úloha: Napíšte program ktorý dekomprimuje vstupný text. Ten je zložený len z veľkých a malých písmen abecedy, znakov [,] a číslíc, pričom sekvencia číslo[text] znamená

číslo opakování textu `text`. Tieto sekvencie môžu byť aj vnorené. Výstup programu má byť dekomprimovaný text zložený len z malých a veľkých písmen.

Príklad:

Vstup:

5[A3[BC]D]

Výstup:

ABCBBCBCDABCBBCBCDABCBBCBCDABCBBCBCD

4. Zimný spánok

15 bodov

Slony sa koncom decembra odoberajú na zaslúžený zimný spánok. Slon Boris nie je v tomto ohľade žiadnou výnimkou. A takisto, ako každý poriadny slon, aj Boris má zo všetkého (jedla) najradšej čerešne.

Vyberie sa teda do jaskyne (v zime predsa čerešne na stromoch nerastú), aby si nazbieral trochu svojej obľúbenej dobroty. Boris je už ospalý a vie, že sa mu neoplatí vracieť sa z jaskyne. Rozhodne sa teda, že v jaskyni pekne prespí do jari. Predtým by ale chcel nazbierať čo najviac čerešní, z ktorých sa predsa tááák sladko spí ...

Úloha:

Jaskyňa má jeden vchod a niekoľko miestností. Tie sú pospájané chodbami, po ktorých Boris môže ísť len v jednom smere. V každej miestnosti je niekoľko čerešní. Vašou úlohou je napísať Borisovi cestu, na ktorej môže nazbierať najviac čerešní. Na vstupe bude počet miestností N . Nasledovať bude N riadkov – popisov jednotlivých miestností. V i -tom riadku budú dve čísla – počet čerešní c_i a počet vychádzajúcich chodieb m_i – a m_i čísel miestností, do ktorých je možné sa týmito chodbami dostať. Boris sa môže po týchto chodbách hýbať len v smere **z** miestnosti i . Vchod má číslo 1 a nie sú v ňom žiadne čerešne. Navyše môžete predpokladať, že ak sa Boris vie nejako (aj cez iné miestnosti) dostať z miestnosti a do miestnosti b , určite sa nevie dostať z miestnosti b do miestnosti a . Na výstupe bude popis cesty – čísla miestností, cez ktoré keď Boris postupne pôjde, nazbiera najviac čerešní. Ak je takýchto ciest viac, vypíšte ľubovoľnú z nich.

Príklad:

Vstup:

9	3 1 7
0 3 4 2 9	2 1 7
5 2 3 7	1 1 8
2 0	5 0
4 2 5 6	6 0

Výstup:

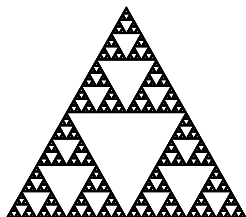
1 4 5 7 8 (Nazbiera tak 13 čerešní, čo je viac, ako keby šiel ľubovoľnou inou cestou.)

5. Z blochou

15 bodov

Už ste počuli o blche Polovičke? Raz sa chela dostať od jednej steny izby k druhej. Darmo jej Mišo vravel, že ak preskočí vždy iba polovicu zostávajúcej vzdialenosti, nikdy sa k druhej stene izby nedostane. Blcha mu oponovala a tvrdila, že to bez problémov zvládne. Po pár skokoch však s dlhým nosom doskákala... O chvíľu tu však bola znovu (prezývali ju tiež Raketa); tvrdila, že keď bude skákať tak, že sa vždy otočí k nejakému rohu izby a skočí do polovice cesty k nemu, síce sa nedostane do žiadneho rohu, ani k stene, ale dostane sa na ľubovoľné miesto v izbe. A tak ju Mišo zobral do miestnosti v tvare trojuholníka a vyzval ju: „Keď si taká úžasná skokanka, taxkoč semka,“ a postavil sa do stredu. A Raketa ostala opäť so spadnutými gambami. Zaujímavé je, kam **všade** dokáže Raketa doskákať a kam nie. Mišo si to začal kresliť a bol celkom prekvapkaný z toho, čo mu vyšlo. Chvíľu sa s tým hral a zistil, že sa tak dajú celkom dobre generovať fraktály (skúste si to aj vy; napíšte si program, ktorý vykreslí tri zadané body trojuholníka; keď má nejaký bod (začne v jednom z rohov), náhodne si jeden roh vyberie a nakreslí bodku do stredu medzi týmto bodom a rohom a pokračuje odtiaľ).

Podme sa teda teraz pozrieť na to, ako sa také fraktály dajú generovať. Fraktály majú tak peknú, oku lahodiacu vlastnosť, že keď ich zväčšíme (prípadne trochu natočíme a posunieme), dostaneme opäť pôvodný fraktál. Príkladom je Sierpinského trojuholník (obr. vľavo).



Všimnite si, že takýto trojuholník sa dá rozložiť na tri o polovicu menšie trojuholníky. Obrázok by sa teda mohol kresliť tak, že nakreslíme trojuholník, rozdelíme ho na tri polovičné. Na tieto tri opäť použijeme tento postup—rozdelíme ich na tri polovičné atď. Matematicky sa toto zmenšovanie, otáčanie a posúvanie robí cez transformácie; keď máme nejaký bod (x, y) , vypočítame nové

$$x' = ax + by + c \qquad y' = dx + ey + f$$

pre nejaké konštanty a, b, c, d, e, f . Dajme tomu, že chceme zobrazíť celý Sierpinského trojuholník na ľavú dolnú časť—všetkým bodom sa potom zmenšia súradnice na polovicu; ostatné dva trojuholníky treba po zmenšení ešte trochu posunúť; tak môžeme dostať takéto transformácie:

(T_1)	$x' = x/2$	$y' = y/2$	pre ľavý dolný trojuholník
(T_2)	$x' = x/2 + 100$	$y' = y/2$	pre pravý dolný trojuholník
(T_3)	$x' = x/2 + 50$	$y' = y/2 + 86.6$	pre horný trojuholník

Keď si vytvoríme takéto transformácie, môžeme vykonať takýto postup: začneme bodom (x_0, y_0) , ktorý patrí do fraktálu; náhodne si zvolíme jednu transformáciu a pomocou nej vypočítame ďalší bod. Takto postupne vypočítame body (x_1, y_1) z (x_0, y_0) , (x_2, y_2) z bodu (x_1, y_1) a tak ďalej. Týmto postupom môžeme ľahko vykresliť nejaký fraktál.

Vyskúšajte napr. zmeniť transformáciu ľavého dolného trojuholníka z vyššie uvedeného príkladu na

(T'_1)	$x' = 0.39x - 0.07y$	$y' = 0.07x + 0.39y$	pre ľavý dolný trojuholník
----------	----------------------	----------------------	----------------------------

Dostanete trochu pokrivený Sierpinského trojuholník.

Na rozbehnutie si ukážeme ešte jeden známy fraktál—Barnsleyovu papraď (obr. vpravo).

Tú môžeme rozdeliť takto: pravá vetva, ľavá vetva a zvyšok paprade (vrch; ten sa opäť skladá z ľavej a pravej vetvy atď.)—aby bola papraď spojená, treba ešte stonku. Ako nájdeme príslušné konštanty? Bez znalostí matic asi najskôr tak, že si nakreslíme čo máme a čo chceme dostať a potom koeficienty budú riešením nejakej sústavy rovníc. Máme šesť koeficientov, teda potrebujeme aspoň šesť rovníc, teda aspoň tri body. Zoštrótime a dostaneme niečo takéto:

(T_1)	$x' = 0.85x + 0.04y$	$y' = -0.04x + 0.85y + 80$	vrch
(T_2)	$x' = -0.15x + 0.28y$	$y' = 0.26x + 0.24y + 22$	pravá vetva
(T_3)	$x' = 0.2x - 0.26y$	$y' = 0.23x + 0.22y + 80$	ľavá vetva
(T_4)	$x' = 0$	$y' = 0.16y$	stonka

Úloha: Napíšte program, ktorý bude používať vyššie popísanú metódu náhodného iterovania bodu a bude kresliť fraktály. Pohrajte sa trochu s konštantami a spolu s popisom programu a zdrojovým kódom nám pošlite niekoľko obrázkov fraktálov (s popísanými transformáciami), ktoré ste pomocou tohoto programu dokázali nakresliť. Hodnotiť sa budú najmä nájdene fraktály, čím krajšie, tým lepšie.