



Korešpondenčný seminár z programovania
XXI. ročník, 2003/04
Katedra vyučovania informatiky FMFI UK,
Mlynská Dolina, 842 48 Bratislava

KSP finančne podporuje MICROSTEP-MIS spol. s r.o.

Príklady 2. kola zimnej časti

Milé naše riešiteľky, milí naši riešitelia,

Zima sa pomaly blíži, hoci to tak teraz nevyzerá. Pre istotu pomaly už brúste hrany na svojich zjazdovkách, kúpte si vosky a teplé spodky z Makyty Púchov. Snáď aj napadne sneh a bude sa dať lyžovať. Keď vás už budú bolieť z lyžovania nohy, sadnite si a pustite sa do riešenia týchto príkladov. Svoje riešenie nám posielajte do **7. januára 2004**. A ako už býva zvykom, nezabudnite na známky v hodnote 15 Sk.

Prajeme vám pekného Mikuláša, Vianoce a Nový rok.

KSPáci

1. O povrchovej ťažbe zlata

15 bodov

Santa a Banta už omrzelo ručné kopanie zlata s krompáčom v tmavej štôlni, a tak si na Vyšnej Klondike založili vlastnú zlatokopeckú spoločnosť. Rozhodli sa pre povrchovú ťažbu bagrami. Nakúpili bagre značky Caterpillar a nechali si vypracovať drahý geologický prieskum.

Geologický prieskum je mapa rozdelená na $N \times M$ tzv. jutroštvorcov. Každý zodpovedá pochopiteľne jednému jutru¹. Pre každý jutroštvorec mapa hovorí, aká je hodnota zlata v dolároch obsiahnutá pod jeho povrchom.

Santovi a Bantovi ostáva už iba odkúpiť od mesta pozemok na ťažbu. Mesto predáva pozemky po d dolárov za jutro, ale nepredá len tak hocijaký kus pozemku. Totiž, z technicko-administratívnych príčin sa zakazuje drobenie jutroštvorcov a navyše sa požaduje, aby pozemok bol tvaru obdĺžnika. Pomôžte Santovi a Bantovi kúpiť vhodný pozemok maximalizujúci zisk, kde zisk z pozemku je rozdiel hodnoty zlata na ňom a jeho ceny za odkúpenie od mesta.

Úloha: Je dané kladné číslo d – cena za jedno jutro, rozmery mapy N, M a mapa $N \times M$ nezáporných čísel, ktoré určujú hodnotu zlata v jednotlivých jutroštvorcoch. Napíšte program, ktorý nájde na mape pozemok tvaru obdĺžnika, z ktorého bude najväčší zisk, a vypíše súradnice jeho dvoch protiľahlých rohov. (Ľavý horný roh mapy má súradnice $[1, 1]$, pravý dolný $[N, M]$.)

Príklad:

Vstup:

$d = 6, N = 3, M = 4$

1 2 0 4

7 5 7 8

1 9 7 6

Výstup:

Najlepšie je kúpiť pozemok s ľavým horným jutroštvorcom so súradnicami $[2, 2]$ a pravým dolným jutroštvorcom so súradnicami $[3, 4]$. Zisk bude 6 dolárov.

2. O plnom prasiatku

15 bodov

Možno ste sa už dočítali, budú sa rušiť desať- a dvadsaťhalierniky. A aj keď o tom ešte neviete, vie o tom Kleofáš, ktorý si už dlšie obdobie dopisuje s Jimmym, kamarátom z USA. Len čo Jimmy pochopil z Kleofášovej lámanej angličtiny, čo sa na Slovensku chystá, až ho

¹asi 5755 m²

striaslo pri predstave, že by sa niečo podobné stalo v jeho rodných Štátoch. To by bolo jeho úsilie našetriť si peniažky na poriadny klavír úplne márne. V panike rozbil hlinené prasiatko, roztriedil si mince a hybaj do hudobní. A že paniku chytil poriadnu, chcel pri kúpe klavíra minúť čo najviac mincí.

Úloha: Jimmy mal v prasiatku K kusov mincí v hodnote 1 cent (penny), L kusov päťcentových mincí (nickel), M kusov desaťcentových mincí (dime) a N mincí v hodnote 25 centov (quarter). Zapáčil sa mu klavír, ktorého cena² je S centov. Poradte Jimmymu, akým najväčším počtom mincí dokáže zaplatiť túto sumu.

Na vstupe sú postupne čísla K , L , M , N a S .

Príklad:

Vstup:

$K = 6, L = 5, M = 4, N = 3, S = 47$

$K = 1, L = 2, M = 3, N = 4, S = 74$

Výstup:

Klavír sa dá zaplatiť najviac 9 mincami.

Klavír sa nedá zaplatiť.

3. O trpaslíkoch a diaľnici

15 bodov

V trpasličej dedine sa chýli k voľbám. Terajší starosta si uvedomil, že sa márne bude snažiť o znovuzvolenie, dokiaľ neotvorí ani jediný úsek diaľnice. Už teraz mu je jasné, že diaľnica by mala viesť cez les, lebo inde už nie je miesto. Kvôli stavbe však nesmú vyťať ani jeden strom, tie majú trpaslíci v úcte. Vôbec nie je dôležité, či diaľnica bude spájať nejaké dôležité miesta, môže sa tiahnuť ľubovoľným smerom, hlavne aby bola. A aby bola rovná, lebo rovné diaľnice sa stavajú rýchlo. A aby bola dosť dlhá, nech sa môže starosta chváliť. Stačí, keď bude dvojprúdová (jeden jazdný pruh v každom smere), aj tak sa po nej asi nebude jazdiť. A najdôležitejšie je, aby jej stredová čiara prechádzala miestom, kde má starostov hlavný protikandidát postavenú chatu.³

Úloha: Na vstupe je dané N – počet stromov v lese a číslo D – šírka jedného jazdného pruhu (teda polovica šírky diaľnice). Nasledujú súradnice N bodov v rovine predstavujúcich stromy. Chata starostovho hlavného konkurenta vo voľbách sa nachádza v bode $[0, 0]$.

Na výstup by mal váš program vypísať, či sa za týchto podmienok dá postaviť diaľnica. Diaľnica má v každom smere od chaty dĺžku väčšiu ako je vzdialenosť najvzdialenejšieho stromu. (Dalo by sa priam povedať, že sa tiahne do nekonečna.)

Vstup:

$N = 3, D = 2.00$

0.00 1.00

1.00 -1.00

-1.00 -1.00

Výstup:

Nedá sa postaviť.

$N = 3, D = 0.80$

0.00 1.00

1.00 -1.00

-1.00 -1.00

Dá sa postaviť.

4. O slovnej hádanke

15 bodov

Miško s Jankom sa pretekajú, kto má lepšiu slovnú zásobu. Lenže všetky hry ich už omrzeli, tak si Miško vymyslel novú. Na jej vysvetlenie si musíme povedať, čo je to pole koncov⁴ ľubovoľného slova.

²samozrejme po zľave :-) ale pre nás to nie je dôležité

³ktorá bude musieť byť pri tej príležitosti zbúraná :-)

⁴Po anglicky sa to zvykne nazývať suffix array.

Napišeme pod seba najprv celé slovo, potom celé slovo bez prvého písmena, potom bez druhého. . . a nakoniec iba posledné písmeno. Tieto riadky budeme volať konce slova. Poprehadzujeme konce slova medzi sebou tak, aby pri čítaní zhora nadol boli v abecednom poradí a očísľujeme ich zhora nadol od 1. Teraz tie konce aj spolu so svojim číslom poprehadzujeme naspäť od najdlhšieho po najkratší. Keď teraz prečítame zaradom iba čísla, dostaneme pole koncov. Napríklad pre slovo kingdzaba vznikla postupnosť čísel 7 6 8 5 4 9 2 3 1.

Priebeh hry je úplne jednoduchý. Miško si myslí slovo, vypočíta pole koncov a napíše ho na papier. Jankovou úlohou je nájsť slovo, ktoré si Miško myslel. Pomôžte Jankovi. Keďže pre dané pole koncov existuje veľa vhodných slov, nájdite také, ktoré má najmenej rôznych písmen.

Úloha: Na vstupe je dané pole koncov ukončené -1. Vašou úlohou je nájsť slovo w , ktoré má najmenší možný počet rôznych písmen a jeho pole koncov je rovnaké ako to na vstupe.

Vstup:

6 2 5 4 3 1 -1

7 6 8 5 4 9 2 3 1 -1

Výstup:

cabbba

dcecbfaba

5. O mravcoch II

15 bodov

Určite si ešte pamätáte, že v prvej sérii dostali mravce na narodeniny kráľovnej počítače. Tieto počítače sú rovnaké a je ich veľmi veľa. Vašou úlohou bude napísať program, ktorý keď nahráme na všetky počítače a naraz ho na všetkých spustíme, niečo zmysluplné spočíta.

Ak si pamätáte, ako sa mravčie počítače programujú, môžete preskočiť priamo k zadaniu úlohy. Pre ostatných teraz bližšie popíšeme, ako vyzerá programovací jazyk, v ktorom sa programujú mravčie počítače. Program budeme písať v jazyku podobnom jazyku Pascal. Premenné použité v programe budú lokálne, t.j. každý počítač bude mať vlastné premenné. Jedinou výnimkou je globálne pole celých čísel `Mem[]`. Toto pole je indexované od 0 a neobmedzene veľké. Na začiatku je v ňom (väčšinou v prvých niekoľko políčkach) uložený vstup úlohy, ostatné jeho políčka obsahujú nuly. Na konci v ňom má ostať uložený výsledok úlohy.

Programy budú spustené naraz na všetkých počítačoch. Počítače budú očíslované od 1 do K . Výpočet bude prebiehať v taktach. Všetky počítače sú rovnako rýchle, v každom takte každý z nich vykoná práve 1 inštrukciu.

Na začiatku programu má byť deklarácia lokálnych premenných, nasledujú samotné inštrukcie. V programe musí každá inštrukcia byť uvedená na samostatnom riadku (teda v jednom takte vykoná každý počítač práve jeden riadok programu). Riadok môžeme ukončiť bodkočiarkou, ňou zároveň začína komentár. Povolene inštrukcie sú:

- priradenie
- prázdna inštrukcia **nop**
- inštrukcia ukončenia výpočtu **halt**
- príkaz skoku **goto** *návestie*
- podmienený príkaz **if** *podmienka* **then** *inštrukcia* (**else** *inštrukcia*)

Pravú stranu priradenia môže predstavovať ľubovoľný aritmetický výraz, podmienkou môže byť ľubovoľný logický výraz. Okrem premenných a konštánt môžu tieto výrazy obsahovať volanie funkcie **cpuid**, ktorá vráti číslo počítača, na ktorom program beží. (Všimnite si, že nemáte k dispozícii kľúčové slová **begin** a **end**.) Návestia sa definujú rovnako ako v Pascale: napísaním *názov*: pred príslušný riadok.

Zostáva vyriešiť prácu so spoločnou pamäťou. Môže sa stať, že niekoľko počítačov chce v tom istom takte čítať z, resp. zapisovať na to isté pamäťové miesto. V takomto prípade najskôr prebehne čítanie, potom zápis. Čítať jedno políčko globálnej pamäte môže naraz ľubovoľne veľa počítačov, ale zapisovať dovoľíme len jednému. Ak by naraz chcelo na to isté miesto zapisovať viac počítačov, program vráti chybu. (T.j. tomuto sa pri písaní programu treba vyhnúť.)

Ukážeme si teraz, v čom je sila mravčích počítačov. Na klasickom počítači potrebujeme na nájdenie najväčšieho z daných N čísel čas $O(N)$. Pozrime sa, ako rýchlo ho vedia nájsť mravce. Nech teda v $\text{Mem}[0]$ je uložené N a v $\text{Mem}[1]$ až $\text{Mem}[N]$ jednotlivé čísla. Spustíme na N počítačoch nasledovný program:

```

var N,num1,num2 : integer;
N:=Mem[0];
1: if (2*cpuid-1 > N) then halt;
   num1:=Mem[2*cpuid-1];
   if (2*cpuid <= N) then num2:=Mem[2*cpuid] else num2:=num1;
   if (num1<num2) then Mem[cpuid]:=num1 else Mem[cpuid]:=num2;
   N:=(N+1) div 2;
   if (N=1) then halt;
   goto 1

```

Po skončení bude v $\text{Mem}[1]$ uložené najväčšie spomedzi daných čísel. Prečo? Počas prvého prechodu cyklom sme rozdelili zadané čísla na dvojice, z každej sme vybrali to väčšie a následne sme „vítazov“ uložili na políčka $1 \dots \lceil N/2 \rceil$ – tým sme vlastne dostali pôvodnú úlohu s približne polovičným počtom čísel, to celé v konštantnom čase! Preto náš program naozaj nájde maximum spomedzi daných N čísel, a to v čase $O(\log N)$.

Úloha: V poli $\text{Mem}[]$ je uložená postupnosť **navzájom rôznych** celých čísel. (V $\text{Mem}[0]$ je uložená jej dĺžka N , v $\text{Mem}[1]$ až $\text{Mem}[N]$ sú jednotlivé prvky postupnosti.) Označme X hodnotu uloženú v $\text{Mem}[1]$. Váš program by mal prvky tejto postupnosti preusporiadať tak, aby obsahovala najskôr (v ľubovoľnom poradí) prvky menšie ako X , potom X a následne prvky väčšie ako X .

Teda ak je na vstupe postupnosť $(5,3,7,1,12,9)$,⁵ jedným z možných výstupov je postupnosť $(3,1,5,9,7,12)$.

Môžete uviesť, koľko počítačov má byť na začiatku spustených v závislosti od N , počet počítačov však smie od N závisieť nanajvýš polynomiálne (teda $N^4 + 7$ počítačov je v poriadku, ale 2^N už nie.)

Prvoradým kritériom hodnotenia bude čas výpočtu vášho programu, nasleduje použitá pamäť (súčet lokálnych pamätí spustených počítačov a počtu použitých políček globálnej pamäte) a počet použitých počítačov.

⁵Teda v $\text{Mem}[0]$ je 6 – dĺžka postupnosti, v $\text{Mem}[1]$ je 5, v $\text{Mem}[2]$ je 3, atď.