Zadania krajského kola súťaže ZENIT v programovaní (1.12.2011)

Na rozdiel od školského kola, na krajskom a celoštátnom máte pre každú úlohu naprogramovať samostatný program. Úlohy budú odovzdávané, testované a vyhodnocované úplne osobitne.

Programy musia čítať vstup zo štandardného vstupu (klávesnice) a písať výstup na štandardný výstup (obrazovku). Programy nesmúotvárať žiadne súbory. V prípade používania delphi alebo iných zložitých editorov treba programovať konzolovú aplikáciu a pred submitom odstrániť prípadné neštandardné knižnice, používané týmto vývojovým prostredím.

Ak správne vyberiete programovací jazyk, potom môže mať súbor ľubovoľné meno a koncovku. Ak má súbor koncovku pas,c,cc,cpp, potom sa dá zvoliť možnosť, aby testovať sám určil programovací jazyk. V prípade riešení v Delphi (dpr súbor) treba vybrať jazyk Pascal.

Po odovzdaní sa súbor na testovači skúsi skompilovať príslušným kompilátorom. V prípade, že sa váš program podarí úspešne skompilovať, testovač ho spustí na vzorových vstupoch, ktoré nájdete v zadaní. Výsledok tohto testu sa vám v testovači po krátkej dobe sprístupní. Z tohto výsledku môžete zistiť, či správne načítavate vstup a či správne vypisujete výstup. Testuje sa na rovnosť vášho výstupu so vzorovým znak po znaku! To znamená, že musíte dodržať presne formát výstupu, nesmiete mať vo výstupe medzery ani žiadne iné znaky navyše a váš výstup musí končiť práve jedným znakom newline (viď vzorový výstup). To, že váš program nevyrieši úspešne niektorý/é zo príkladov, neznamená, že nemôže dostať žiadne body pri ostrom testovaní. Ostré testovanie prebehne po súťaži na celkom iných vstupoch. Testovaný bude posledný odovzdaný program.

Časový limit je, ak sa nepovie inak, 200 milisekúnd. Testovač je rýchly stroj, takže ak vám program na lokálnom počítači trvá na veľkom vstupe beží pár sekúnd, malo y to byť v pohode. Pamäťový limit: 64 MB. Limit na veľkosť programu: 100 kB. Ďalšie detaily na stránke v sekcii Pomoc.

Možné odpovede testovača: OK, Time Limit Exceeded (váš program neskončil v požadovanom časovom limite), Wrong Answer (výstup vašeho programu sa nezhodoval znak po znaku so správnym), Runtime Exception (počas vyhodnocovania váš program vykonal chybu a spadol, alebo požaduje príliš veľa pamäte), Security Exception (to isté ako Runtime Exception).

O učiteľovi, ktorý bol posadnutý päťkami

V tejto sade úloh sa budeme venovať nešťastnému učiteľovi, ktorý trpí nenapraviteľnou náklonnosťou k číslu 5. Ako uvidíte, život takéhoto človeka nie je vôbec jednoduchý a preto ste boli požiadaní, aby ste nášmu smutnému hrdinovi pomohli a zjemnili mu jeho utrpenie. Ak nie je uvedené ináč, časový limit je 200 milisekúnd (testovač je dosť rýchly stroj, takže ak vám program na najväčšom vstupe beží na vašom súťažnom počítači aj pár sekúnd, stále by mal byť dosť rýchly).

a) (10 bodov) Napíšte program, ktory nebude čítať žiadny vstup a vypíše kúzelné heslo “Plače Jožko, plače Betka, keď v zošite svieti päťka!” (úvodzovky len pre prehľadnosť, nie sú súčasť výstupu, vypisujte bez diakritiky.)

b) (15 bodov) Náš učiteľ zarába tak veľa peňazí, že nevie čo s nimi. Rozhodol sa preto, že nadbytočné úspory, ktoré sa mu nezmestia vo forme mincí do veľkého trezoru na kopci, pokukladá na terminovaný účet. Raz sa tak doma na svojom stole pozrel na zmluvu a na nej na číslo svojho účtu a zistil, že medzi ciframi tohto čísla sa vyskytuje príliš málo pätiek. Zo zúfalstva túto zmluvu roztrhal na márne kúsky. Po čase ho ale zlosť prešla a rád by znova zistil číslo svojho účtu. Prečítajte zo vstupu **N** (2 <= **N** <= 4) - počet fragmentov čísla, ktoré našiel na zemi na rôznych kusoch papiera, a vypíšte všetky možnosti,

ako mohlo toto číslo účtu vyzerať. Možné čísla vypisujte na každý riadok jedno a usporiadané rastúco. Číslo účtu má vždy najviac 10 cifier.

Príklad:
Vstup:
2

11 1111

Výstup:

111111

Vstup:

3

78 5 7878

Vystup:

5787878

7857878

7878578

7878785

c) (20 bodov) Učiteľa vždy zaujímalo postavenie päťky medzi ostatnými ciframi. Na úvod by ho zaujímalo, koľko výskytov cifry 5 je vo všetkých **K** ciferných číslach. Prečítajte **K** (1 <= **K** <= 19) a na jediný riadok vypíšte odpoveď. Môžete predpokladať, že odpoveď sa zmestí do 64 bitovej premennej so znamienkom.

Príklad:
Vstup:
2
Výstup:
19

Vstup:
13
Výstup:
11800000000000

d) (25 bodov) Pokračujeme vo vyjasňovaní postavenia cifry 5 vo svete matematiky. Učiteľa odjakživa priťahovali čísla, ktoré pozostávajú len z cifry 5. Keďže ale napriklad sté najväčšie také číslo má sto cifier, ťažko sa zapisuje. Preto vymyslel zaujímavý spôsob – miesto veľkého čísla zo samých pätiek si zapamätáme dve menšie čísla, ktoré v súčine dajú toto číslo. V tejto úlohe vás učiteľ prosí o pomoc so

súvisiacim problémom: pre dané číslo **a** (1 < **a** < 10^15) chceme nájsť najmenšie číslo **b** také, aby súčin **a.b** obsahoval samé '5'. Môžete predpokladať, že výstup nebude mať viac ako 1000 cifier.

Príklad:
Vstup:
3
Výstup:
185

Vstup:
51
Výstup:
1089324618736383442265795206971677559912854030 5

e) (25 bodov) Učiteľ už chcel ísť s kamarátmi na kofolu a futbal, ale napokon si povedal, že vyrieši ešte jednu matematickú úlohu. Táto úloha má jednoduché zadanie. Keďže ale nie je jednoduché ju vyriešiť, pomôžte mu. V tejto úlohe je na vstupe jediné číslo **N** (1 <= **N** <= 1000) a zaujíma nás najmenšie také číslo, ktoré je deliteľné každým číslom, ktoré je menšie rovné **N**.

Príklad:
Vstup:
5

Výstup:

60

Vstup:

57

Výstup:

1642493587250378254392 00

f) (25 bodov) Učiteľ napokon s vašou pomocou zradné úlohy vyriešil. Predtým, ako môže ísť za kamarátmi, musí ale žiakov odviesť na obed a robiť počas neho dozor. Preto by chcel, aby to trvalo čo najkratší čas. To samozrejme závisí od prejdenej vzdialenosti. Prečítajte rozmery mapy školy **R** a **C**, (3 <= **R,C** <= 54) a na následných **R** riadkoch samotnú mapu. Na každom z týchto riadkov je práve **C** znakov, ktoré sú alebo **X** (stena), alebo **.** (bodka, chodba). Okrem týchto znakov je práve jeden znak **T** (trieda) a práve jeden znak **M** (mäso). Na výstup vypíšte jediné číslo – dĺžku najkratšej cesty medzi triedou a jedálňou. Dĺžku cesty meriame v krokoch medzi políčkami, pričom za susedné políčka považujeme tie, medzi ktorými je hrana (políčko má najviac štyroch susedov). Môžete predpokladať, že okrajové rady a stĺpce nákresu sú steny.

Príklad:
Vstup:
4 15

```
XXXXXXXXXXXXXXX
X..X......XXXMX
XX..T.XX......X
XXXXXXXXXXXXXXX
```
Výstup:
12

Vstup:
20 51

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X..................X.............................X
X........X.......T.X....XXXXXXXXXXXXXXXXXXXXXX....X
X.XXXXXXXXXXXXXXXXX....X...................X....X
X..................X....X....XXXXXXXXXXXX....X....X
X..................X....X....X..........XX.XXX....X
XXXXXXXXXXXXXXXXX..X....X....X...................X
X..................X....X....XXXXXXXXXXXXXXXXXXXXXX
X.......XXXXXXXXXXXXXXX.X...................X....X
X..................X....X..XXXXXXXXXXXXXXXXXXXXXXXX
X..................X....X......................X
X..................X....XXXXXXXXXXXXXXXXXXXXXX....X
X.XXXXXXXXXXXXXXXXXX.XXXX...................X....X
X..................X....X....XXXXXXXXXXXX....X....X
X..................X....X....X..M.......XX...X....X
XXXXXXXXXXXXXXXXX..X....X....X..............X....X
X..................X....X....XXXXXXXXXXXXXXXXXXXX.X
X.......XXXXXXXXXXXXXXX.X...................X....X
X......................X......................X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```
Výstup:
275

g) (25 bodov) Učiteľ sa rozhodol obohatiť život svojim žiakom a zobral ich na filmový festival algoritmického programovania. Ako ale vysvitlo, na tomto festivale vznikol vážny problém pri organizácii a učiteľ so žiakmi bol požiadaný o pomoc pri jeho riešení. Organizátorom sa podarilo prenajať dve kinosály v multi extra ultra hyper 4D kinovom komplexe. Do programu bolo zaradených **N** (2 <= **N** <= 8) zaujímavých filmov. Každý z filmov má svoju celočíselnú dĺžku v minutach. Medzi filmami v kinosálach kvôli optimálne využitému času nebudú prestávky. Pre účastníkov je ale najlepšie, ked môžu medzi filmami zmeniť kinosálu, pretože nie každdého zaujíma každý film. Napríklad niektorých účastníkov nezaujímajú memoizácie backtrackov, ale nadovšetko ocenia najnovšie trendy v programovaní prehľadávaní grafov. Organizátori sa teraz pokúšajú zostaviť program v kinosálach – kde a kedy sa ktorý film bude premietať. Celý festival prebehne v jeden deň a

premietanie sa začne v rovnakú minútu v oboch kinosálach. Chceme spraviť harmonogram premietania tak, aby existovalo čo najviac minút takých, že v danú minútu končia filmy v oboch kinosálach (vtedy sa dá pohodlne prejsť z jednej sály do druhej). Vo vstupnom súbore sa na prvom riadku nachádza počet vstupných sád (najviac 5). Každá vstupná sada pozostáva z dvoch riadkov. Na prvom z nich sa nachádza číslo **N** a na ďalšom riadku **N** medzerami oddelených čísel medzi 1 a 100 vrátane – dĺžky filmov v minútach. Pre každú testovaciu sadu vypíšte jeden riadok a na ňom text: **Vstup X: Y**, kde **Y** je odpoveď na príslušný testovací prípad. V tejto úlohe je časový limit 2 sekundy.

Príklad:
Vstup:
3

4

10 20 20 30

6

10 20 20 20 40 30

8
20 30 40 50 60 70 80 10
Vystup:
Vstup 1: 1

Vstup 2: 2

Vstup 3: 2

V prvom príklade sú dve možnosti: do prvej kinosály dať filmy s dĺžkami 10 a 30 (daná minúta bude 40.) alebo filmy s dĺžkami 20 a 10 (daná minúta bude 20., ak v druhej sále pôjde prvý film ten, ktorý má 20 minút).

h) (30 bodov) Na odreagovanie sa po práci pri počítačmi sa učiteľ so žiakmi vybral skrášliť areál školy. Požičali si od školníka lopaty, krompáče, rýle a motyky a presunuli sa pred budovu, kde sa rozhodli zasadiť niekoľko okrasných a niekoľko úžitkových plodín. Keďže ale zlyhala organizácia, žiaci začali zvážať zeminu a kopať jamy rôznych hĺbok úplne nekoordinovane. Keď učiteľ po krátkom čase anarchiu zarazil, vedľa každého žiaka zostala alebo jama, alebo kopec zeminy. Učiteľ sa zamyslel, ako by sa dala táto situácia dať čo najrýchlejšie do poriadku a povedal si, že skúsi najprv nájsť trojicu žiakov tak, aby množstvo naznášanej zeminy touto trojicou presne stačilo na zahladenie jám vykopaných touto trojicou.

Na školskom pozemku pracovalo **N** žiakov (medzi 1 a 1000 vrátane). Každý žiak alebo vykopal jamu, v ktorej chýba niekoľko litrov zeminy, alebo nazvážal niekoľko litrov zeminy (pričom žiaci pracovali tak precízne, že daný objem je vždy celočíselný údaj v litroch). Na vstupe na ďalšom riadku nasleduje **N** celých nenulových čísel (najviac 100,000,000 v absolútnej hodnote), pričom záporné číslo označuje jamu s príslušným objemom a kladné nazvážanú zeminu s príslušným objemom. Nájdite takú trojicu čísel, ktorej súčet bude 0. Môžete predpokladať, že taká existuje. Na výstup vypíšte trojicu indexov čísel zo vstupu, usporiadanú rastúco. Ak je takých viac, vypíšte takú, ktorá má čo najmenšie prvé číslo,

a ak je stále odpoveď nejednoznačná, tak takú, ktorá má druhé číslo čo najmenšie.

Príklad:
Vstup:
6
7 -1 3 4 -2 -1
Výstup:
2 3 5
Vstup:
5
9 -6 -5 -4 -3
Výstup:
1 2 5
Medzi sample vstupmi na testovači sa nachádza aj jeden veľký, ktorý tu ale nebudeme z praktických dôvodov uvádzať.

i) (20 bodov) Na nášho učiteľa prišla sťažnosť na **krajský úrad pre dozor nad činnosťou v školstve, výskume a vede**. Istý dôležitý úradník sa dopočul, že učiteľ údajne používa nespravodlivé hodnotiace metódy a preto ho prišiel do školy vyšetrovať. Prvé, čo chce inšpektor vidieť, sú známky, aké učiteľ udelil svojim žiakom. V tejto úlohe sú známky študenta postupnosti (reťazce) znakov **1,2,3,4,5**. Niektoré sekvencie známok sú podozrivé a upútajú inšpektorovú pozornosť. Napríklad, udeliť žiakovi jednotku, následne päťku a následne dvojku môže naznačovať veľkú nerozhodnosť, nestabilitu a celkovú neschopnosť objektívnej kvalifikácie zo strany učiteľa. Keďže ale politiké časy sú nestabilné, podozrivé postupnosti známok sa menia, upravujú a starostlivo prispôsobujú novým pomerom. Pomôžte učiteľovi a napíšte mu program, ktorý prečíta podozrivé sekvencie známok a potom v jeho klasifikačnom hárku identifikuje študentov, ktorých krajský inšpektor označí za podozrivých. Vo vstupnom súbode je najprv číslo **N** (1 <= **N** <= 1000): počet podozrivých podreťazcov. Nasleduje **N** riadkov, každý s jedným neprázdnym reťazcom znakov **1,2,3,4,5**, dlhým najviac 6 znakov. Na ďalšom riadku vstupného súboru je prirodzené číslo **M**, nepresahujúce 100 – počet klasifikovaných žiakov. Na nasledujúcich **M** riadkoch sú neprázdne reťazce známok, dlhé najviac 50. Vypíšte jediný riadok a v ňom medzerou oddelený zoznam indexov podozrivých žiakov. Tento zoznam by mal byť utriedený vzostupne. V prípade, že známky žiadneho žiaka nie sú podozrivé, nechajte jediný riadok výstupu prázdny.

Príklad:
Vstup:
2
152
3
4
33125
421425
3
5555521
Výstup:
1 3

Vstup:
4

12
13
14
15
3
333
311
113
Výstup:
3

j) (40 bodov) Táto úloha nadväzuje na úlou h). Učiteľ si povedal, že nepripustí žiadne pochybnosti o svojej kompetencii a skúsi sa prispôsobiť systému. Za polrok chce udeliť presne 15 známok. Zaujíma ho, koľko rôznych postupností 15 známok sa môže vyskytnúť tak, aby žiadna nebola pre úrady podozrivá.

Prečítajte zo vstupného súboru číslo **M** (medzi 1 a 1000 vrátane) a na každom z nasledujúcich riadkov neprázdny reťazec znakov 1,2,3,4,5 dlhý najviac 6, popisujúcich podozrivé sekvencie známok. Vypíšte počet bezpečných postupností 15 známok (teda takých, v ktorých sa nevyskytuje žiadna podozrivá sekvencia). V tejto úlohe je časový limit jedna sekunda.

Príklad:
Vstup:
4
1
2
3
4
Výstup:
1
(jediná korektná sekvencia známok pozostáva zo samých pätiek)

Vstup:
10
21
31
32
41
42
43
51
52
53
54
Výstup:
3876
(všetky korektné postupnosti známok sú neklesajúceho tvaru – po známke musí nasledovať rovnaká alebo horšia)