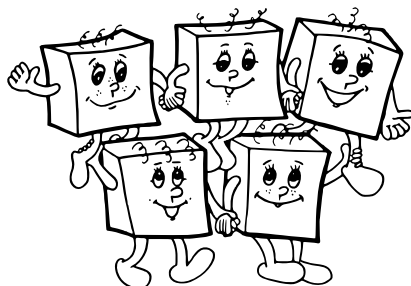


OLYMPIÁDA V INFORMATIKE NA STREDNÝCH ŠKOLÁCH

<http://oi.sk/>



tridsiaty tretí ročník
školský rok 2017/2018

zadania celoštátneho kola, deň 2 **kategória A**

Priebeh celoštátneho kola

Celoštátne kolo 33. ročníka Olympiády v informatike, kategórie A, sa koná v dňoch 21.-24. marca 2018. Na riešenie úloh druhého, praktického dňa majú súťažiaci 4,5 hodiny čistého času. Akékoľvek pomôcky okrem písacích potrieb (napr. knihy, výpisy programov, kalkulačky) sú zakázané.

Čo má obsahovať riešenie úlohy?

- Skompilovateľný program v jazyku C++ alebo Pascal. Ak sa váš program nepodarí na našom testovacom počítači skompilovať, bude automaticky hodnotený 0 bodmi.

Hodnotenie riešení druhého (praktického) dňa

Sú tri úlohy. Ku každej úlohe máme pripravených 10 sád testovacích vstupov. Sada vstupov pozostáva z jedného alebo viacerých testovacích vstupov. Za každú sadu vstupov, ktorej všetky vstupy (každý zvlášť) váš program správne vyrieši, získate jeden bod.

Testovanie na každom vstupe prebieha samostatne. Spustíme váš program a na štandardný vstup mu dáme konkrétne vstupné údaje. Hovoríme, že váš program daný vstup vyriešil, ak splní nasledujúce kritériá:

- Skončí skôr ako uplynie stanovený časový limit.
- Neprekročí stanovený pamäťový limit.
- Skončí korektne, nie chybou počas behu.
- Dáta, ktoré vypíše na štandardný výstup, tvoria korektný výstup, zodpovedajúci danému vstupu.
- Nebude používať žiadne funkcie zakázané kvôli bezpečnosti testovacieho systému.

Počas súťaže môžete priebežne odovzdávať svoje riešenia. Odovzdané riešenie bude otestované a dozviete sa svoj bodový zisk. (V prípade preťaženia testovača môžu organizátori obmedziť toto priebežné testovanie na vhodnú podmnožinu všetkých testovacích dát.)

Po ukončení súťaže zoberieme pre každú úlohu váš posledný odovzdaný program a ten otestujeme na všetkých testovacích vstupoch. Vaše výsledné body za úlohu budú body získané týmto programom.

Sady vstupov sú navrhované tak, aby každé korektné riešenie získalo nejaké body, bez ohľadu na to, ako pomalé je. Bližšie informácie o testovacích dátach nájdete na konci zadania každej úlohy.



A-III-4 Laser

Máš obdĺžnikový stôl. Na povrchu stola je nakreslená štvorcová sieť. Na niektorých políčkach siete, vrátane skoro všetkých na obvode stola, sú prekážky. Na obvode stola sú len dve výnimky. Jednou z nich je políčko, na ktorom je umiestnený laser, namierený dovnútra stola. Druhou z nich je políčko, kde je fotoreceptor: zariadenie, ktoré vie rozpoznať, keď je zasiahnuté laserovým lúčom.

Na niektorých políčkach stola sú už postavené sklenené kocky, ktorými sa nedá hýbať. Kockami laserový lúč prechádza bez akýchkoľvek efektov.

Na voľné políčka stola vieš umiestňovať zrkadlá. Každé zrkadlo je obojstranné (t.j. obe jeho strany odrážajú) a musí byť umiestnené na uhlopriečku políčka. Na voľné políčko vieme teda zrkadlo umiestniť dvoma spôsobmi. Laserový lúč pri náraze do zrkadla zmení svoj smer o 90 stupňov.

Súťažná úloha

Daná je bitmapa predstavujúca súčasný stav stola. Zistite, či sa dá vhodným pridaním zrkadiel priviesť laserový lúč od lasera k fotoreceptoru. Ak áno, zistite, koľko najmenej zrkadiel treba a nájdite jedno vyhovujúce rozmiestnenie optimálneho počtu zrkadiel. (Každé pridané zrkadlo rátame len raz, a to aj keby sa od neho laserový lúč postupne odrazil dvakrát – teda raz od každej jeho strany.)

Formát vstupu a výstupu

V prvom riadku vstupu sú rozmery stola r a s . Zvyšok vstupu tvorí bitmapa popisujúca stôl: r riadkov a v každom z nich s znakov. Znak ‘.’ (bodka) predstavuje voľné políčko, znak ‘X’ prekážku, znak ‘O’ (veľké o) sklenenú kocku, znak ‘-’ laser a znak ‘=’ fotoreceptor.

Je zaručené, že obvod bitmapy tvoria jeden laser, jeden fotoreceptor a okrem nich už len samé prekážky. Je tiež zaručené, že laser aj fotoreceptor sa nachádzajú v prvom alebo v poslednom stĺpci bitmapy a nie sú v jej rohoch. Laser strieľa lúč kolmo na stranu stola, na ktorej leží.

Ak je možné lúč z lasera priviesť do fotoreceptoru, vypíšte r riadkov a v každom z nich s znakov: bitmapu zo vstupu, do ktorej sú doplnené ako zrkadlá znaky ‘/’ a ‘\’. Zrkadlá musia priviesť laserový lúč až do fotoreceptoru a ich počet musí byť najmenší možný. Ak existuje viacero optimálnych riešení, môžete vypísať ľubovoľné z nich. (Aj dĺžka laserového lúča môže byť ľubovoľná, záleží len na počte zrkadiel.)

Ak nie je možné dosiahnuť fotoreceptor, vypíšte namiesto toho riadok s textom „nemozne“.

Obmedzenia a hodnotenie

Je desať sád vstupov. V sádach 2, 4, 6, 8 a 10 hodnoty r a s neprekročia 10, 20, 300, 1500, resp. 2500. Sada $2k - 1$ má rovnaké obmedzenia ako sada $2k$ a navyše platí, že sa v nej nevyskytujú žiadne sklenené kocky. V sádach 1 a 2 navyše platí, že všetky prekážky ležia na obvode stola – teda žiadne nie sú v jeho vnútri.

Príklady

| vstup | výstup | vstup | výstup | vstup | výstup |
|---|--|---|---|---|---------|
| <pre>4 7 XXXXXX XXXXXX X.....= -.....X XXXXXX</pre> | <pre>XXXXXX X../..= -../..X XXXXXX</pre> | <pre>7 7 XXXXXX XXXXXX -.....X XXX.XXX X..O..X X.X.X.X =.X...X XXXXXX</pre> | <pre>XXXXXX -..\.X XXX.XXX X/.O.\X X.X.X.X =/X\./X XXXXXX</pre> | <pre>4 7 XXXXXX X0000- X.....= XXXXXX</pre> | nemozne |

V prvom príklade treba pridať aspoň dve zrkadlá. V druhom príklade ich treba ich treba aspoň šesť – ale keby nám tam nezavadzala tá sklenená kocka, stačili by štyri. V treťom príklade laserový lúč preletí celým druhým riadkom a narazí na prekážku. Keďže tento riadok je plný sklenených kociek, nevieme doň nikam umiestniť zrkadlo.



A-III-5 Graffiti

Pozdĺž nemenovanej štátnej hranice ktosi nedávno postavil dlhोčizný múr. Skladá sa z p panelov, ktoré si pre účely tejto úlohy očísľujeme pozdĺž múru od 1 po p . Keď bol múr čerstvo postavený, bol celý krásne čistý. V budúcnosti však očakávame, že sa budú diať dva typy udalostí:

- Občas príde samozvaný umelec so sprejmi a bude chcieť pomalovať nejaký súvislý úsek čistých panelov.
- Občas príde poriadková služba a vyčistí nejaký súvislý úsek panelov (či už čistých alebo pomalovaných).

Súťažná úloha

Vašou úlohou bude napísať program, ktorý bude takéto udalosti efektívne spracúvať.

O poriadkovej službe sa vždy dozvieme presný úsek panelov, ktorý vyčistí.

O umelcovi sa dozvieme len dĺžku úseku čistých panelov, ktoré chce pomalovať. O umelcoch navyše vieme, že všetci bývajú na začiatku múru a sú leniví. Ak má umelec na výber veľa rôznych dostatočne dlhých čistých úsekov múru, vždy si vyberie ten, ktorý začína panelom s najmenším číslom.

Formát vstupu a výstupu

V prvom riadku vstupu sú dve kladné celé čísla: počet panelov p a počet udalostí u .

Zvyšok vstupu tvorí u riadkov, každý popisuje jednu udalosť v chronologickom poradí. Príchod umelca bude popísaný riadkom tvaru „U d “, kde d je dĺžka úseku, ktorý chce pomalovať. Príchod poriadkovej služby bude popísaný riadkom tvaru „P x y “, kde $x \leq y$ sú čísla prvého a posledného panelu v práve vyčistenom úseku.

Pre každý príchod umelca vypíšete jeden riadok a v ňom jedno číslo: číslo panelu, na ktorom začína úsek, ktorý pomaluje.

Ak sa niekedy stane, že na múre nie je nikde taký dlhý čistý úsek, aký by umelec potreboval, umelec sklamaný odíde domov a nič nepomaluje. V takomto prípade vypíšete číslo -1 .

Obmedzenia a hodnotenie

Je 5 sád testovacích vstupov, za každú môžete získať 2 body.

V prvej sade platí $p \leq 100$ a $u \leq 1000$.

V druhej sade platí $p, u \leq 50\,000$.

V tretej sade platí $p, u \leq 150\,000$.

V štvrtej a piatej sade platí $p \leq 500\,000$ a $u \leq 300\,000$. Vo štvrtej sade navyše pre každého umelca platí $d = 1$.

Vo všetkých sádach v každej udalosti platí $1 \leq d \leq 10^9$ a $1 \leq x \leq y \leq p$.

Príklad

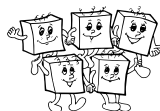
vstup

```
10 5
U 6
P 2 4
U 5
U 4
U 2
U 47
```

výstup

```
1
-1
7
2
-1
```

Prvý umelec pomaluje panely 1 až 6. Poriadková služba vyčistí panely 2 až 4. Druhý umelec by chcel pomalovať päť panelov, nikde však práve nie je päť čistých panelov vedľa seba. Tretí umelec pomaluje panely 7 až 10 – to je v danej chvíli jediná štvorica čistých panelov vedľa seba. Štvrtý umelec pomaluje panely 2 a 3. Piaty umelec by toho chcel pomalovať viac ako má celý múr, má teda smolu.



A-III-6 Korálky

Ešte nedávno mala Natálka krásny náhrdelník, na ktorom bolo n korálok. Korálky mali rôzne odtiene ružovej, postupne prechádzali z tmavej do svetlej. No náhrdelník sa jej pretrhol a korálky sa rozkotúľali po zemi.

Natálka teraz drží prázdnu šnúрку a Peťko (ktorý sa na pretrhnutí náhrdelníka nechtiac podieľal) kľáči na zemi, zbiera korálky a jednu po druhej ich Natálke podáva. Tá môže každú korálku buď navliecť na šnúрку (a to z ľubovoľného konca), alebo ju môže odložiť do krabice s šitím.

Natálke pri zbieraní korálok napadla nasledovná úloha: Čo keby korálky skúsila navliekať tak, aby na konci boli na šnúрке zoradené od najtmavšej po najsvetlejšiu? Možno sa jej nepodarí takto navliecť všetky korálky, ale určite bude zábavné skúsiť ich navliecť čo najviac.

Súťažná úloha

Na vstupe je daná postupnosť f_1, \dots, f_n : farby korálok v poradí, v akom ich Peťko podával Natálke. Všetky f_i sú **navzájom rôzne** prirodzené čísla. Čím väčšie číslo, tým svetlejšiu korálku predstavuje.

Pre každú korálku si Natálka vybrala jednu z troch možností: buď ju navliekla na ľavý koniec šnúrky, alebo ju navliekla na pravý koniec šnúrky, alebo ju odložila preč. Vieme, že na konci celého procesu mala šnúрку, na ktorej zľava doprava farba korálok prechádzala z tmavej do svetlej – teda im zodpovedajúce čísla rástli.

Napište program, ktorý vypočíta, koľko najviac korálok mohla mať nakoniec Natálka na šnúрке.

Formát vstupu a výstupu

V prvom riadku vstupu je kladné celé číslo n . V druhom riadku je n rôznych kladných celých čísel f_1, \dots, f_n . Vypíšte jeden riadok a v ňom jedno číslo: maximálny počet korálok, ktoré mohli skončiť na šnúрке.

Obmedzenia a hodnotenie

Je 10 sád testovacích vstupov, za každú môžete získať 1 bod. Vo všetkých sádach pre každé i platí $1 \leq f_i \leq 10^9$. Jednotlivé sady majú rôznu maximálnu hodnotu n : 6, 12, 20, 60, 100, 5 000, 17 000, 150 000, 250 000 a 500 000.

Príklady

| vstup | výstup |
|---|--------------------------------|
| <input type="text" value="5"/> <input type="text" value="30 10 50 20 40"/> | <input type="text" value="3"/> |

Jedno optimálne riešenie: Korálku 30 navlečie na šnúрку (je jedno z ktorej strany), korálku 10 nepoužije, korálku 50 navlečie sprava, korálku 20 navlečie zľava a korálku 40 nepoužije.

| vstup | výstup |
|--|--------------------------------|
| <input type="text" value="6"/> <input type="text" value="10 20 9 21 8 22"/> | <input type="text" value="6"/> |
| <input type="text" value="7"/> <input type="text" value="1 7 4 3 5 2 6"/> | <input type="text" value="5"/> |

Tu je optimálne nepoužiť prvé dve korálky a vyrobiť na šnúрке postupnosť 2, 3, 4, 5, 6.

TRIDSIATY TRETÍ ROČNÍK OLYMPIÁDY V INFORMATIKE

Príprava úloh: Michal Anderle, Eduard Batmendijn, Michal Forišek, Samuel Gurský, Samuel Sládek, Emanuel Tesař

Recenzia: Michal Forišek

Slovenská komisia Olympiády v informatike

Vydal: IUVENTA – Slovenský inštitút mládeže, Bratislava 2018