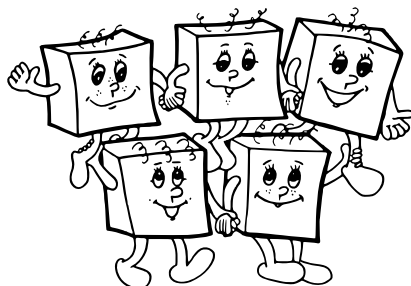


OLYMPIÁDA V INFORMATIKE NA STREDNÝCH ŠKOLÁCH

<http://oi.sk/>



tridsiaty prvý ročník
školský rok 2015/2016

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

Priebeh celoštátneho kola

Celoštátne kolo 31. ročníka Olympiády v informatike, kategórie A, sa koná v dňoch 6. – 9. apríla 2016. 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.

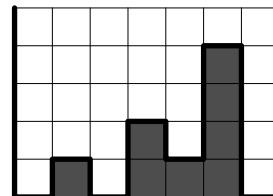
Sady vstupov sú navrhované tak, aby každé korektné riešenie získalo nejaké body, bez ohľadu na to, ako pomalé je. Bližšie informácie o testovacích dátach nájdete na konci zadania každej úlohy.



A-III-4 Kaňon

V dvojrozmernom svete objavili nedávno prieskumníci novú atrakciu: obrovský kaňon. Príklad takéhoto kaňonu vidíte na obrázku vpravo. Kaňon na obrázku má šírku $n = 7$. Jeho dno môžeme popísať postupnosťou výšok: $(0, 1, 0, 2, 1, 4, 0)$. Každé číslo udáva, koľko štvorcíkov zeme sa v danom stĺpci nachádza. Naľavo a napravo od krajných stĺpcov sú steny kaňonu. Tie sú zvislé a dostatočne vysoké.

Nad kaňonom občas prší. Zaujímavé je, že keď prší, zásadne prší len v niektorom jednom stĺpci. Voda z oblakov vždy padá rýchlosťou 1 štvorcík za sekundu. (T.j., za každú sekundu dažďa padne toľko vody, koľko zaplní jeden štvorcík na obrázku kaňonu.) Voda, ktorá dopadne na dno kaňonu, sa spojito rozlieva do strán, z vyšších miest na nižšie.



Súťažná úloha

Daný je popis kaňonu. Pre každý jeho stĺpec zodpovedzte nasledujúcu otázku: ak v tomto stĺpci začne pršať, po koľkých sekundách začne v tomto stĺpci stúpať vodná hladina?

(Rozmyslite si, že odpoveď na túto otázku nezávisí od toho, ako presne sa voda dovtedy rozlievala po okolí. Voda v konkrétnom stĺpci začne stúpať vtedy, keď už zaplnila všetky nižšie položené časti kaňonu v okolí.)

Formát vstupu a výstupu

V prvom riadku vstupu je kladné celé číslo n .

V druhom riadku je n nezáporných celých čísel a_1, \dots, a_n : popis dna kaňonu zľava doprava.

Na výstup vypíšte jeden riadok a v ňom postupnosť n celých čísel: pre každý stĺpec kaňonu zľava doprava odpoveď na vyššie uvedenú otázku. (Medzi každými dvoma číslami vypíšte jednu medzeru. Bezprostredne za posledným číslom vypíšte znak konca riadku.)

Obmedzenia

V sadách #1 a #2 platí $n \leq 100$ a $a_i \leq 100$.

V sadách #3 a #4 platí $n \leq 1000$ a $a_i \leq 10^6$.

V sadách #5 až #10 platí $a_i \leq 10^9$.

V sadách #5 až #10 hodnota n postupne rastie od $n = 100\,000$ v sade #5 po $n = 2\,000\,000$ v sade #10.

V sadách s nepárnym číslom sú všetky a_i navzájom rôzne.

Príklady

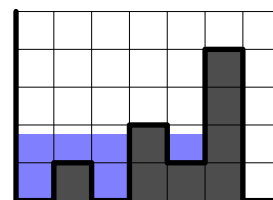
vstup

```
7
0 1 0 2 1 4 0
```

výstup

```
0 2 0 6 0 20 0
```

Príklad vstupu zodpovedá kaňonu na obrázku na začiatku zadania. Na obrázku vpravo je jeden možný príklad toho, ako môže tento kaňon vyzeráť po tom ako 5 sekúnd pršalo v prostrednom stĺpci. O ďalšiu sekundu by sa všetky okolité dolinky zaplnili vodou úplne a potom by už začala pribúdať voda aj v prostrednom stĺpci.



vstup

```
8
0 2 0 1 0 0 1 0
```

výstup

```
0 12 0 4 0 0 4 0
```

Pozor na to, čo sa deje v prípade, že viaceré úseky dna kaňonu majú presne rovnakú výšku. Keď napríklad prší vo štvrtom stĺpci, bude po 3 sekundách po jednom štvorcíku vody v treťom, piatom a šiestom stĺpci. Následne počas štvrtej sekundy začne pretekať voda ponad siedmy stĺpec do ôsmeho. Po štyroch sekundách už bude súvislá vodná hladina siahať od tretieho až po ôsmy stĺpec. A ak by teraz pršalo ďalej, bude voda stúpať naraz na celom tomto úseku, a teda už aj v štvrtom stĺpci.



A-III-5 Demonštrácia

V hlavnom meste Absurdistanu sa chystá demonštrácia proti prílišnému množstvu demonštrácií. Počas demonštrácie sa chce dav protestujúcich občanov presunúť z námestia pred sídlo veľkého vezíra. To však nie je také jednoduché ako sa môže zdať. Hlavný problém je v tom, že dav ľudí nemôžete len tak presúvať ulicami: ak sa pokúsíte zo širokej ulice vojsť do úzkej, hrozí vám, že sa vo vzniknutej tlačnici mnohí ľudia zrania.

Súťažná úloha

Daný je popis mesta: počet n križovatiek, počet m ulíc medzi nimi a šírka každej ulice. Každá ulica je jednosmerná a spája nejaké dve križovatky (v Absurdistane nemôžete ísť proti smeru jednosmerky, ani keď ste chodec).

Ďalej sú dané križovatky a, b odkiaľ a kam treba dav dostať. Ako posledné je dané nezáporné číslo k s nasledovným významom: ak dav ide ulicou šírky s , v najbližšej križovatke môže prejsť len na ulicu, ktorej šírka je aspoň $s - k$.

Vašou úlohou je napísať program, ktorý zistí, či je vôbec možné dostať dav z križovatky a na križovatkú b . Ak áno, zistíte, cez koľko najmenej ulíc pri tom dav musí prejsť.

Formát vstupu a výstupu

V prvom riadku vstupu je päť nezáporných celých čísel: n, m, a, b, k . Ich význam je vysvetlený vyššie. Križovatky sú očíslované od 0 po $n - 1$. Nasleduje m riadkov, každý popisuje jednu ulicu. Popis každej ulice tvoria tri celé čísla x_i, y_i, s_i : číslo križovatky, v ktorej daná ulica začína, číslo križovatky kde ulica končí, a šírka tejto ulice. Vypíšte jeden riadok a v ňom jedno celé číslo: najmenšie d , pre ktoré existuje riešenie, pri ktorom dav postupne prejde d ulicami. Ak neexistuje žiadny spôsob ako dostať dav z a do b , vypíšte číslo -1 .

Obmedzenia

Vo všetkých sadách platí: $2 \leq n, 0 \leq m$ (sú aspoň 2 križovatky a 0 ulíc), $0 \leq a, b < n, a \neq b$ (štart a cieľ sú korektné a navzájom rôzne), $0 \leq x_i, y_i < n, x_i \neq y_i$ (čísla križovatiek sú korektné a žiadna ulica nemá oba konce na tej istej križovatke), $0 < s_i$ (ulice majú kladné šírky).

V prvých troch sadách platí $n, m, k, s_i \leq 100$.

V sadách #4 až #6 platí $n \leq 100\,000, m \leq 500\,000, k, s_i \leq 20$.

V sadách #7 až #10 platí $n \leq 100\,000, m \leq 500\,000, k, s_i \leq 10^9$.

V sadách #7 a #8 navyše platí, že v každej križovatke sa stretá nanajvýš 20 ulíc.

Príklady

vstup

```
3 2 0 2 7
0 1 20
1 2 10
```

výstup

-1

Máme 3 križovatky a 2 ulice. Chceme ísť z križovatky 0 na križovatkú 2. Konštanta k je 7.

Prvá ulica ide z križovatky 0 na križovatkú 1 a má šírku 20. Druhá ulica ide z 1 na 2 a má šírku 10. Rozdiel medzi šírkami ulíc je priveľký, dav ktorý príde prvou ulicou nevie pokračovať druhou, preto riešenie neexistuje.

vstup

```
4 4 0 2 7
0 1 20
1 2 10
1 3 14
3 1 9
```

výstup

4

Dav postupne prevedieme prvou, treťou, štvrtou a druhou ulicou. Všimnite si, že križovatkou 1 prejdeme dvakrát.



A-III-6 Obdĺžnikový tetris

Roman by si chcel naprogramovať hru Tetris. Programovanie mu však zatiaľ príliš nejde, preto sa rozhodol, že si najskôr spraví nejakú zjednodušenú verziu. V prvom rade v jeho verzii nikdy nič nebude miznúť. Keď raz nejaký dielik niekam dopadne, ostane tam celý až do konca hry. V druhom rade sa nič nebude otáčať ani posúvať. V ktorých stĺpcoch sa dielik zjaví, v tých bude padať dodola, až kým na niečo nenarazí. A v treťom rade, dieliky rôznych nepravidelných tvarov sú príliš zložité. V Romanovej hre budú padať len samé obdĺžniky.

Súťažná úloha

Hrací plán (tzv. šachta) je obdĺžnik, ktorý má šírku s a výšku 10^9 . Na začiatku hry je celý hrací plán prázdny. Riadky aj stĺpce šachty čísloujeme od 0 (stĺpce zľava doprava, riadky zdola hore). Počas hry sa postupne, jeden po druhom, zjaví na vrchu šachty n obdĺžnikov. V poradí i -ty spomedzi nich bude mať šírku w_i políčok, výšku h_i políčok a najľavejší stĺpec, v ktorom tento obdĺžnik leží, bude stĺpec ℓ_i .

Napište čo najefektívnejší program, ktorý bude simulovať padanie týchto obdĺžnikov.

Formát vstupu a výstupu

V prvom riadku vstupu je šírka šachty s a počet obdĺžnikov n . V každom z nasledujúcich n riadkov sú parametre w_i , h_i a ℓ_i jedného z obdĺžnikov.

Pre každý obdĺžnik vypíšte jeden riadok a v ňom jedno celé číslo: číslo riadku, v ktorom skončí spodok dotyčného obdĺžnika.

Obmedzenia

V prvých troch sadách platí $s \leq 100$, $n \leq 100$ a súčet všetkých h_i neprekročí 100.

Vo štvartej a piatej sade platí $s \leq 1000$ a $n \leq 100\,000$.

V ďalších sadách sa hodnoty s a n postupne zväčšujú, pričom v poslednej sade platí $s \leq 2\,000\,000$ a $n \leq 1\,000\,000$.

Vo všetkých testovacích vstupoch platí $1 \leq w_i$, $\ell_i + w_i \leq s$ a $1 \leq h_i \leq 1000$.

(Obdĺžniky majú kladné rozmery, žiaden netrčí bokom zo šachty a všetky dokopy sa určite zmestia do šachty.)

Príklad

vstup

10	7
1	1 4
4	2 3
2	2 8
2	2 6
2	2 8
1	5 2
10	1 0

výstup

0
1
0
3
2
0
5

obrázok
Obsah šachty na konci tejto ukázkovej hry:

