



Korešpondenčný seminár z programovania

Leták letnej časti XLI. ročníka

Korešpondenčný seminár z programovania (KSP) je súťaž programátorov – stredoškolákov a mladších – pripravovaná skupinou študentov Fakulty matematiky, fyziky a informatiky Univerzity Komenského v Bratislave. Naším cieľom je zdokonaľiť žiakov v programovaní a v algoritmickom myslení.

Riešením súťažných úloh a štúdiom vzorových riešení sa zlepšíš v programovaní a naučíš sa algoritmicky rozmýšľať. Získané poznatky a skúsenosti využiješ v iných súťažiach v programovaní (napríklad pri riešení [Olympiády v informatike](#)), v bežnom živote, počas vysokoškolského štúdia, dokonca aj na prijímacích pohovoroch do zamestnania. Naši riešitelia sa každoročne zúčastňujú a úspešne umiestňujú na medzinárodných olympiádach v informatike (v Austrálii, Taliansku, Kazachstane, Taiwane, ...). Mnoho našich bývalých riešiteľov sa tiež bez ťažkostí zamestnalo v špičkových IT spoločnostiach ako Google, Facebook, ESET, ...

Ak študuješ na strednej škole a zaujíma ťa programovanie, neváhaj a zapoj sa do KSP:

Ako sa zapojiť do KSP?

- **Prečítaj** si zadania. Nájdeš ich v tomto letáku a na našej stránke <https://www.ksp.sk/ulohy>. Každý rok máme zimnú a letnú časť, obe majú dve kolá s ôsmimi úlohami.
- Teš sa, aké sú tento rok pekné úlohy.
- **Vyrieš** úlohy. Nemusíš vyriešiť všetky, nemusíš ich vyriešiť najlepšie ako sa dá. Aj za čiastočné riešenia sa dostávajú body, za každú úlohu za dá získať 0 až 20 bodov.
- Na riešenie úloh jedného kola máš približne dva mesiace a môžeš ich riešiť doma bez toho, aby si niekam cestoval. Termín odovzdania úloh je napísaný aj na našej stránke, aj v PDF zadaniach. Úlohy sa nedajú odovzdávať po termíne, takže si to, prosím, nenechaj na poslednú chvíľu.
- Úlohy rieš samostatne a neprehrádzaj riešenia ostatným riešiteľom. Odpisovanie riešení a prezradenie riešení pred termínom kola je porušením pravidiel KSP. Po skončení kola sa, samozrejme, o riešeníach rozprávať môžeš. :)
- **Odozdaj** riešenia úloh. Odkaz na odovzdávanie úloh nájdeš pod webovým zadaním každej úlohy alebo na stránke <https://www.ksp.sk/odovzdavanie>. Na odovzdávanie sa treba prihlásiť, aby sme vedeli, komu máme dať body.
 - Vo väčšine úloh odovzdávaš program a popis.
 - Program je hneď po odovzdaní otestovaný testovačom a hneď vidíš, koľko bodov za program máš. Program môžeš odovzdávať znova a znova, až kým nie si spokojný s výsledkom. Ak nevieš, ako majú vyzeráť odovzdané programy, pozri si <https://www.ksp.sk/odovzdavanie-programov>
 - Do popisu slovne napíšeš, ako tvoje riešenie funguje, prečo funguje a tiež odhad časovej a pamätovej zložitosti programu. Viac sa dozvieš na stránke <https://www.ksp.sk/ako-riesit>. Popis opraví a obodujú vedúci KSP po skončení kola.
- Po skončení kola si **prečítaj vzorové riešenia** úloh (veľa sa z toho naučíš), pozri svoje opravené popisy (či ti tam vedúci nenapísali nejaké poučné komentáre), pozri sa do výsledkovky a **teš sa**, koľko máš bodov. Vo výsledkoch sa hodnotí samostatne letná a zimná časť. V každej časti je dôležitý celkový súčet bodov.
- Prečo sa máš tešiť z bodov? Čítaj ďalej.

Čo môžem vyhrať?

- Okrem neoceniteľných vedomostí, skúseností a zručností, ktoré získaš pri riešení semináru, môžeš vyhrať množstvo skvelých vecí.
- Všetci víťazi od nás dostanú **vecné ceny**.
- Pre 36 najlepších riešiteľov organizujeme každoročne dve týždenné **sústredenia**. Sústredenie je niečo ako tábor, na ktorom spoznáš nových priateľov s podobnými záujmami, naučíš sa čosi viac nielen o programovaní a zažiješ kopec zábavy. Sústredenia sú fakt skvelé akcie, najmä, keď ich organizuje Trojsten.

- Aby ste sa mohli pochváliť ostatným, akí ste šikovní, víťazom všetkých levelov udelíme a pošleme **diplomy**.
- Aj keď sa nedostaneš medzi víťazov, stále môžeš byť úspešným riešiteľom. Úspešný riešiteľ je ten, kto získal aspoň polovicu bodov počas celej časti (letnej, či zimnej).

Pravidlá a levely

Počnúc tridsiatym piatym ročníkom rušíme staré kategórie a prechádzame na nový systém *levelov*.

Každý riešiteľ má level, číslo od 1 po 4. Noví riešitelia začínajú na leveli 1 a pokiaľ sa im v riešení darí, level im postupne rastie. Svoj level si môže každý riešiteľ pozrieť na našej stránke. Riešiteľom s levelom L sa započítavajú body len za úlohy s číslami L až 8.

Vo výsledkových listinách (<https://www.ksp.sk/vysledky>) sa každému riešiteľovi počíta **5 najlepšie vyriešených úloh**. Celkovo sa dá za časť (dve kolá) získať 200 bodov. Riešitelia, ktorí sa v nejakej výsledkovke umiestnili na jednom z prvých dvoch miest a majú aspoň 150 bodov sú **víťazi**. Najlepších 36 riešiteľov pozývame na sústredenie.

Podrobnejšie pravidlá si môžete prečítať na <https://www.ksp.sk/pravidla>.

Registrácia

Pred odovzdaním riešenia je potrebné sa zaregistrovať na našej webstránke a vyplniť požadované kontaktné údaje. Odporúčame sa zaregistrovať aspoň pár dní pred odovzdávaním riešenia (pre prípad, že by ste mali počas registrácie nejaké problémy).

Účastou v KSP nám dávate súhlas spracovať a archivovať údaje, ktoré nám poskytnete pri registrácii, ako aj zverejniť vaše meno, školu, ročník a získané body vo výsledkovej listine.



Úlohy 1. kola letnej časti

Termín odoslania riešení tohto kola je pondelok **15. apríla 2024**. Doprogramovávanie končí v piatok 26. apríla 2024.

1. Lúka nešťastia

12 b za popis, 8 b za program

Pastier Larry má problém. Kolmo cez pasienku kde sa pasú jeho ovečky prechádza prekliata rieka nekonečného zatratenia a bolesti. Ovečku, ktorá by do nej omylom padla čaká neistý ale nepredstaviteľne mrzký osud. Navyše strata akejkoľvek ovečky prináša pre chudobného pastiera značnú ekonomickú škodu. Larry má ale nápad ako túto hrozbu mitigovať.

V stodole má k dispozícii niekoľko veľkých obdĺžnikových oceľových plátov. Pripevní ich cez rieku, aby po nich mohli ovečky bezpečne prechádzať na druhú stranu. Pasienka má členitý terén a rieka má nerovné brehy, takže pláty nemôže namontovať kam len chce, ale našiel nejaké potenciálne miesta kde sa to dá. Všetky majú rovnakú veľkosť a otočenie (sú rovnobežné s riekou).

Čím viac plátov pastier Larry rozmiestni, tým menej ovečiek stratí. Ale pláty sa nesmú prekryvať, aby sa nenarušila ich stabilita. Poradte pastierovi Larryho, koľko najviac plátov môže cez rieku pripevniť.

Úloha

Pôdorys pasienky je nekonečná 2D rovina. Rieku si môžeme predstaviť ako priamku $y = 0$.¹ Na pasienke je n osov zarovnaných obdĺžnikov, čo sú miesta kde Larry môže potenciálne umiestniť plát. Všetky majú rovnakú šírku s a výšku v a pretínajú rieku.

Vyber na ktoré miesta má Larry namontovať oceľový plát a na ktoré nie, tak aby ich bolo čo najviac ale žiadne vybraté miesta sa neprekryvali. Dotýkať sa môžu.

Formát vstupu

V prvom riadku vstupu je číslo n udávajúce počet potenciálnych obdĺžnikov.

V druhom riadku je prirodzené číslo s a v treťom riadku prirodzené číslo v , určujúce šírku a výšku každého obdĺžnika.

Nasleduje n riadkov ktoré popisujú jednotlivé potenciálne miesta. V každom sú dve celé čísla x a y , čo predstavuje obdĺžnik ktorého štyri rohy sú (x, y) , $(x + s, y)$, $(x, y + v)$ a $(x + s, y + v)$. Všetky obdĺžniky pretínajú rieku, čiže $y < 0 < y + v$. Všetky obdĺžniky na vstupe sú navzájom rôzne. Obdĺžniky sú **zoradené vzostupne** podľa ich x -ovej súradnice. Ak majú viaceré rovnaké x , môžu byť medzi sebou zoradené akokoľvek.

Sú 4 sady vstupov, v ktorých platia tieto obmedzenia:

Sada	1	2	3	4
$1 \leq n \leq$	10	100	1 000	10 000
$1 \leq s \leq$	10	100	100 000	100 000
$2 \leq v \leq$	10	100	100 000	100 000
$ x \leq$	10	1 000	1 000 000	1 000 000

Formát výstupu

Vypíš jedno číslo: veľkosť najväčšej podmnožiny vzájomne sa neprekrywajúcich obdĺžnikov.

¹Vlastne je to skôr taký potôčik, ale "rieka" znie vznešenejšie.

Príklady

vstup

```
4
2
5
-5 -4
-4 -3
-1 -1
1 -2
```

výstup

```
3
```

Prvé 2 obdĺžniky sa prekrývajú, preto musíme vybrať práve jeden z nich. Je jedno ktorý. Ostatné sa s nijakým obdĺžnikom neprekrývajú, iba dotýkajú, takže ich vyberieme oba. Maximum je teda 3.

2. Elity filmového plátna

12 b za popis, 8 b za program

V krajine Absurdistan prekvitá filmový priemysel. A ako sa patrí, každý rok sa koná veľká slávnostná ceremónia, kde sa zídu všetci herci z krajiny a tí najlepší dostanú ceny. Ceny sa udeľujú z absurdného množstva kategórií ako “najžiarivejší úsmev”, “najtragickejší monológ” alebo “najdivokejší výkrik”. Môže to dopadnúť všelijako. Už sa doteraz párkrát stalo, že kategórii bolo viac ako hercov a každému hercovi sa ušla aspoň jedna cena, ale raz aj že jediná ultra-populárna herečka vyhrala úplne všetky kategórie.

Ceremónia prebieha tak, že zaradom v každej kategórii moderátor prečíta meno víťazného herca, potrasie mu rukou, odovzdá mu zlatú sošku a všetci zatlieskajú. Moderátor samozrejme tiež musí byť nejaký herec, lebo diváci majú radi keď je to známa tvár a nie nejaký neznámy človek zo zákulisia. Každá kategória má práve jedného moderátora a jedného víťaza. Medzi kategóriami sa moderátori môžu striedať.

Úloha

Ste členmi organizačného výboru tohtoročnej ceremónie. Zoznam a poradie kategórií už je pevne daný a porota už vybrala všetkých víťazov. Vašou úlohou je už len naplánovať, kto bude kedy robiť moderátora. Na to máte dve dôležité podmienky.

1. V žiadnej kategórii nesmie byť moderátor a víťaz ten istý človek. Nevyzeralo by dobre, keby odovzdával sošku sám sebe, diváci by si mysleli že je to nejaký podvod. Ktorýkoľvek iný herec môže kategóriu moderovať, len nie víťaz.
2. Je lepšie aby sa moderátori nestriedali príliš často. Lebo vždy keď sa striedajú, ten starý bude strašne dlho rečniť že prichádza nový moderátor, a ten nový moderátor mu bude celú večnosť ďakovať za privítanie. A to dokonca aj keby ten nový už niečo predtým moderoval. Čas sú peniaze a divákov to hrozne nudí, takže sa snažte aby viaceré kategórie za sebou moderoval ten istý herec.

Nemusíte vypísať detaily kto presne bude kedy moderovať, stačí keď vypíšete koľko najmenej striedaní moderátorov musí nastať. Prvú kategóriu môže moderovať ľubovoľný herec (okrem jej víťaza), jeho nastúpenie sa neráta ako striedanie.

Formát vstupu

Na prvom riadku je číslo t - počet testovacích vstupov (rôznych ceremónií). Každú ceremóniu vyhodnoťte zvlášť.

V rámci každej ceremónie je na prvom riadku číslo h - počet hercov. Na ďalších h riadkoch sú mená jednotlivých hercov v nejakom neurčitom poradí. Mená majú maximálne 100 znakov a môžu obsahovať ľubovoľné normálne znaky (viditeľné ASCII a medzery). Žiadni dvaja herci sa nevolajú rovnako.

Na ďalšom riadku je číslo k - počet kategórií. Na ďalších k riadkoch sú mená jednotlivých víťazov (v prvej, druhej, ..., k -tej vyhlásenej kategórii). Každé meno víťaza je naozaj niekto z tých h hercov čo sa zúčastnili tejto ceremónie.

V jednotlivých sadách platia nasledujúce obmedzenia:

Sada	1	2
$1 \leq t \leq$	20	20
$2 \leq h \leq$	1 000	10^5

Sada	1	2
$0 \leq k \leq$	1 000	10^6
Dĺžka celého vstupu \leq	1 MiB	2 MiB

Formát výstupu

O každej z t ceremónii vypíšte jeden riadok s jedným číslom: minimálny počet striedaní moderátorov medzi kategóriami.

Príklad

vstup

```
3
5
Jozef Mrkva
Marina Inoue
Tatiana Lieskovoorieskova
Karol XVI. Gustav
GEORG
10
Jozef Mrkva
Jozef Mrkva
GEORG
Karol XVI. Gustav
GEORG
Marina Inoue
Karol XVI. Gustav
Marina Inoue
Tatiana Lieskovoorieskova
GEORG
3
Adam
Božena
Cyril
5
Božena
Božena
Božena
Cyril
Božena
2
Alica
Bob
7
Alica
Bob
Alica
Bob
Alica
Bob
Alica
```

výstup

```
1
0
6
```

Prvú ceremóniu môže začať Tatiana Lieskovoorieskova, vyhrala však deviatu kategóriu, takže ju potom musí niekto prestriedať. Druhú ceremóniu vie celú bez striedania odmoderovať Adam. V tretej ceremónii sa budú Bob s Alicou medzi každou kategóriou striedať.

3. Gigantický Legoland

12 b za popis, 8 b za program

Vitajte, vitajte! Počula som, že v Legolande ste prvý raz. Nevadí, vysvetlím vám, ako to tu funguje. Je to

vskutku celkom jednoduché. Vidíte? Tam pred vami v jednom rade sú všetky atrakcie, ktoré tu máme. Určite ich chcete všetky navštíviť. Ale počkať, ešte som neskončila. Čo prosím? Že zem je hrbolatá? To nie, to len stojíte na legokocke. Lahko viete skočiť na inú. Samozrejme, skočiť môžete len na kocku, ktorá je hneď vľavo, vpravo, hore alebo dole voči vašej momentálnej kocke. Takto si budete hopkať aj medzi našimi atrakciami. A teda, medzi dvoma atrakciami vždy prejdete Manhattanskú vzdialenosť. A teraz k pravidlám vašej návštevy. Sú len dve. Prvé: atrakcie, ktoré navštívite musia tvoriť súvislý úsek. Druhé: (pre dobro rodičov, ktorí sa snažia prejsť z atrakcie A do atrakcie B, bez toho, aby prešli okolo atrakcie C (vraj to ušetrí veľa detských slz), pre každú trojicu z úseku musí platiť, že súčet vzdialeností medzi dvoma dvojicami sa nemôže rovnať vzdialenosti medzi tretou dvojicou. To je celé. Jednoduché, však? Vyberte si úsek atrakcií, ktoré chcete navštíviť a do toho! Počet úsekov, z ktorých si môžete vybrať je... Vlastne, nemôžem robiť všetkú robotu za vás. Určite na to zvládnete prísť sami.

Úloha

Dané je pole R s n atrakciami: r_1, r_2, \dots, r_n . Súradnice i -tej atrakcie sú (i, r_i) . Vzdialenosť medzi atrakciou $a = (a, r_a)$ a atrakciou $b = (b, r_b)$ je: $v(a, b) = |a - b| + |r_a - r_b|$. Vašou úlohou je nájsť počet súvislých úsekov pre ktoré platí, že v nich **neexistuje** trojica atrakcií i, j, k , pre ktoré platí $v(i, j) = v(i, k) + v(k, j)$.

Formát vstupu

V prvom riadku vstupu je číslo n udávajúce počet atrakcií v poli. V prvej sade platí $1 \leq n \leq 30$ a v druhej sade platí $1 \leq n \leq 2 \cdot 10^5$. V nasledujúcom riadku je n čísel reprezentujúcich ypsilonové súradnice atrakcií r : r_1, r_2, \dots, r_n . Pre obe sady platí $1 \leq r_i \leq 10^9$.

Formát výstupu

Vypíš jeden riadok a v ňom jedno celé číslo: počet súvislých úsekov, ktoré môžete v Legolande navštíviť.

Príklad

vstup

```
4
3 5 2 4
```

výstup

```
10
```

V prvom prípade je jedno, ktorý úsek vyberiete. Všetky sú dobré. Že prejdeme okolo tretej. Všetky súvislé úseky sú dobré.

vstup

```
5
7 10 2 10 7
```

výstup

```
12
```

V druhom prípade si môžete všimnúť, že okolo úseku $[r_1, r_2, r_3, r_4]$ nemôžete prejsť, lebo by sa nám mohlo stať, že pri prechode od prvej atrakcie k štvrtej by ste prešli okolo druhej.

4. Odporné psiská, FUJ!

12 b za popis, 8 b za program

Dog Show Winner² má veľmi rád svojho krásneho KSPsíka a vyhrávanie psích prehliadok. Čo však nemá rád, sú iné psy. Ustavične očuchávajú jeho miláčika. Nanešťastie pre neho sú psy v jeho rodnom meste veľmi obľúbené – sú doslova na každej ulici! Dog Show Winner je mimo iného veľmi excentrický. Každéj ulici priradil číslo, podľa toho, ako veľmi bude jeho miláčik očuchaný, keď po nej prejde. A keďže je fakt veľmi excentrický, tak zisťuje ako veľmi bol jeho pes očuchaný bitovým *OR*-om. Na to aby mohol vyhrávať psie prehliadky, potrebuje chodiť na súťaže, ktoré sú všade v meste. Potrebuje teda nájsť také ulice v meste, aby sa pomocou nich vedel dostať kamkoľvek a zároveň aby bol jeho havko očuchaný čo najmenej. Sám to ale nezvládne a potrebuje vašu pomoc.

Úloha

Mesto reprezentujeme ako graf s ováňovanými hranami. Vašou úlohou je vybrať také hrany, aby graf zostal súvislý a jeho *OR* bol najmenší možný. Medzi dvomi vrcholmi vedie najviac jedna hrana. Graf je vo všetkých vstupoch súvislý.

²<https://www.lego.com/en-my/kids/characters/minifigures/dog-show-winner-7dbda2a81f7a42378edad062167df3d2>

Formát vstupu

Na prvom riadku vstupu dostanete dve čísla – počet vrcholov v a počet hrán e . Nasleduje e riadkov po troch číslach. V i -tom riadku sú čísla x_i, y_i a w_i – vrcholy, ktoré spája i -ta hrana a jej váha. Vrcholy indexujeme od 0.

V sadách platia nasledovné obmedzenia:

Sada	1	2	3	4
$2 \leq v \leq$	10	1 000	10^4	10^5
$1 \leq e \leq$	19	1 000	10^4	10^5
$0 \leq \max w_i \leq$	1 000	10^4	10^6	$2 * 10^9$

Formát výstupu

Vypíšte jedno číslo – najmenší možný bitový OR kostry grafu zadaného na vstupe.

Príklady

vstup

```
4 5
1 0 10
0 2 4
0 3 4
3 1 4
2 1 7
```

výstup

```
4
```

Keď si zoberieme hrany medzi 0 a 2, 0 a 3, 3 a 1 dostaneme súvislý graf s najmenším možným bitovým OR-om.

vstup

```
2 1
0 1 3
```

výstup

```
3
```

Musíme zobrať všetky hrany v grafe (čiže tú jednu), aby bol súvislý.

5. LTT

12 b za popis, 8 b za program

Boli ste neposlušní darebáci a za trest vás odsúdili do *LEGO® Tunela Trýznenia™*. Tento tunel je dlhočížny a aby toho nebolo málo, jeho podlaha je posiatá LEGO kockami. Jooj LEGO na zemi. Au au.

Spoločnosť LEGO sa aj v Tuneli Trýznenia snaží rozvíjať rozum a plánovanie. Na každom metri tunela sú umiestnené tenučké šlapky. Tie síce nijako nezmiernia bolesť spôsobenú dánskym plastom, ale dovidia vám robiť kroky väčšej dĺžky. Konkrétne, ak máte na nohách šlapky s veľkosťou v , môžete urobiť krok dĺžky v metrov.

Netušíte ako je to možné, neviete prečo takéto šlapky nepredávajú armáde a bojíte sa pýtať. Ale to nie je dôležité. Dôležité je, že vám to umožní prejsť tunelom na menej krokov, a teda aj menej bolesti.

A tak teraz stojíte bosí na začiatku tunela, vedľa vás položené tenučké šlapky. Hľadíte do tunela a premýšľate. Kde sa prezúvať? Koľko krokov treba spraviť? Čo bude na večeru?

Úloha

Tunel je reprezentovaný ako postupnosť n čísel v_0, v_1, \dots, v_{n-1} , kde v_i je veľkosť šlapiek na i -tom metri tunela. Na začiatku stojíte na nultom metri tunela a vašou úlohou je vyjsť von z tunela (teda stúpiť kamkoľvek do vzdialenosti aspoň n) na najmenší možný počet krokov. Na každom metri tunela môžete urobiť krok dĺžky v metrov, ak máte na nohách šlapky veľkosti v . Na začiatku nemáte na nohách žiadne šlapky. Môžete si šlapky prezúvať kedykoľvek tak, že šlapky ktoré máte (alebo nemáte) na nohách vymeníte s tými, na ktorých stojíte.

V popise nezabudnite napísať zložitosti a ich poriadne dôkazy. Riešenia s časovou zložitou $O(n^2)$ alebo horšou nebudú za plný počet bodov.

Formát vstupu

V prvom riadku vstupu je číslo n ($1 \leq n \leq 20\,000$) a slovo *smer* udávajúce počet šlapiek v tuneli a dovolený smer pohybu. Ak je *smer* rovný **one**, môžete sa pohybovať len dopredu, ak je *smer* rovný **two**, môžete sa pohybovať aj dozadu (ale nesmiete stúpiť na zápornú súradnicu).

V druhom riadku je postupnosť n čísel v_0, v_1, \dots, v_{n-1} ($1 \leq v_i \leq n$) udávajúca veľkosti šlapiek na jednotlivých metroch tunela.

V jednotlivých sadách platia nasledujúce obmedzenia:

Sada	1	2	3	4
$1 \leq n \leq$	10	2000	20 000	20 000
<i>směr</i>	one	one	one	two

Formát výstupu

Vypíšte jeden riadok a v ňom jedno celé číslo – najmenší počet krokov, ktorými môžete prejsť za koniec tunela.

Príklad

vstup	výstup
<pre>6 one 1 2 4 1 2 1</pre>	<pre>3</pre>

Na začiatku sa prezujeme do šlapiek 1, potom spravíme dva kroky a prezujeme sa do šlapiek 4 a už nám stačí len jeden krok do ciela. Keby sme sa prezuli do šlapiek 2, museli by sme urobiť štyri kroky (lebo nestačí stúpiť na posledné šlapky).

Tipové okienko

Pripomíname, že je odporúčané Python kód nemať v globálnom skópe, ale zabaliť ho do funkcie³.

Dnešný nový tip je zase pre používateľov jazyka C++. Ak používate dynamické štruktúry, oplatí sa ich inicializovať na začiatku na veľkosť, ktorú budete potrebovať. Funkcia `reserve` je prítomná nielen na `std::vector`, ale aj `std::unordered_map` a iných.

V podobnom duchu sa toto tiež oplatí použiť keď pracujeme napríklad s dynamickým intervalovým stromom. Tento prístup sa volá `bump allocator` a funguje rovnako ako `reserve` s `std::vector`. Na začiatku si rezervujeme veľkú časť pamäte pre `Node structy` a potom si ich postupne berieme ako nám treba. Toto nám ušetrí veľa alokácii pri volaní `new`.

Nie vždy je toto nutné robiť, ale ak sú alokácie *bottleneck*, vieme touto metódou dosiahnuť veľké zrýchlenie.

6. Aaaaaaaa!

12 b za popis, 8 b za program

V Legolande býva aj Indiana Jones, známy archeológ a akčný hrdina. Všetci s ním radi trávajú čas, pretože vedľa, že priam priťahuje dobrodružstvo. Nikoho teda neprekvapilo, keď sa jedného dňa Indy pri prechádzke po ulici prepadol cez zatarasený vchod do starodávnej jaskyne.

Indyho to samozrejme veľmi potešilo, najmä preto, že na stene okamžite zbadal mapu jaskyne, ktorá celkom jasne ukazuje, že na samom spodku sa nachádza obrovský poklad. Rád by sa k nemu čo najrýchlejšie dostal (a to, či sa potom dostane aj von, ho momentálne vôbec nezaujímá).

Keďže však jaskyňa našiel náhodou (cestou na nákup), nemá so sebou svoj bič, ani žiadne iné vybavenie, okrem masívneho krompáča (pochopiteľne). Takže keď raz niekam v jaskyni zoskočí, nevie už vyliezť vyššie. Môže niektoré steny jaskyne rozbiť, aby sa dostal ďalej, ale kopať s krompáčom je namáhavé (dokonca viac, než ho nosiť všade so sebou), takže by sa tomu chcel čo najviac vyhnúť.

Pomôžte Indymu naplánovať cestu tak, aby sa dostal na spodok jaskyne a pritom musel čo najmenejkrát použiť krompáč. A ak sa tam dostať nedokáže, povedzte mu, nech tam radšej nelezí.

Úloha

Jaskyňa, ktorú Indy objavil, má tvar mriežky s výškou V a šírkou S . Niektoré jej políčka sú priechodné, ostatné obsahujú kocky. Kocky síce Indy dokáže rozbiť, ale stojí ho to veľa času. Preto by chcel nájsť takú cestu nadol, ktorú dokáže prejsť s čo najmenším počtom rozbitých kociek. Vôbec mu pritom nevedí, že prejde veľa políčok, keďže chodenie je omnoho menej namáhavé, než rozbíjanie.

Indy začína v ľavom vrchnom rohu jaskyne. Pohybovať sa dokáže len o jedno políčko doprava alebo doľava, ak sa nachádza nad prázdny políčkou, okamžite spadne. Počas cesty nikdy nesmie spadnúť z väčšej výšky, než P políčok, inak si ublíži.

³Viac info v minulom trikovom okienku <https://www.ksp.sk/ulohy/zadania/2612/>

Rozbíjať dokáže Indy len kocky, ktoré sa od neho nachádzajú o jeden riadok nižšie, o jedno políčko vedľa a políčka nad nimi sú voľné. V príkladoch nižšie sú všetky takéto kocky označené X, ostatné kocky #, prázdne políčka . a Indyho aktuálna poloha I.

```
..I..  .#I..  ..I..  ##I##
#X#X#  ###X#  #.#X#  ..###
```

Vašou úlohou je vypísať minimálny počet kociek, ktoré musí Indy rozbiť, aby sa dostal na spodok (do ľubovoľného políčka spodného riadku) jaskyne bez toho, aby si ublížil pádom. Ak sa dolu nevie dostať žiadnym spôsobom, vypíšete “Do tejto jaskyne sa liezt neoplati”.

Formát vstupu

V prvom riadku sú čísla V , S a P oddelené medzerou. Platí, že $P \leq V$. Nasleduje V riadkov dlhých S znakov – znaky môžu byť . (prázdne políčko) alebo # (kocka). V ľavom hornom políčku tabuľky je vždy prázdne políčko, pod ním je vždy kocka.

Každá zo štyroch testovacích sád je za dva body. V jednotlivých sádach platia nasledovné obmedzenia:

Sada	1	2	3	4
$2 \leq V \leq$	10	20	50	100
$2 \leq S \leq$	10	20	60	100
$1 \leq P \leq$	10	20	50	100

V prvej a tretej sade zároveň platí $P = V$, teda Indy nikdy nespadne z tak veľkej výšky, aby sa mu niečo stalo.

Formát výstupu

Vypíšete jeden riadok obsahujúci buď číslo (najmenší možný počet kociek, ktoré musí Indy rozbiť, aby sa dostal do spodného riadku jaskyne), alebo reťazec “Do tejto jaskyne sa liezt neoplati”. Nezabudnite vypísať znak konca riadku.

Príklady

vstup

```
5 5 5
...#
###.
####.
...##
.####
```

výstup

2

vstup

```
5 5 1
.....
####.
####.
...##
.####
```

výstup

4

vstup

```
5 5 1
...#
###.
####.
.####
...##
```

výstup

7

vstup

```
5 4 5
...
###.
####
####
.###
```

výstup

```
Do tejto jaskyne sa liezt neoplati
```

V prvom príklade stačí Indymu vykopať kocku na súradniciach $[2,1]$ (súradnice x,y číslujeme od nula od ľavého vrchného rohu), zoskočiť na uvoľnené miesto, spraviť krok doprava, vykopať políčko $[2,2]$ a potom už len dôjsť do cieľa vľavo dole.

V druhom príklade by mohol postupovať podobne a vykopať políčka $[3,1]$, $[2,1]$ a $[2,2]$, ale všimnime si, že potom by spadol z výšky 2, čo by mu ublížilo. Namiesto $[2,2]$ teda musí vykopať $[3,2]$ a potom ešte $[3,3]$.

V treťom príklade môže vykopať $[2,1]$, (potom!) $[1,1]$, ďalej $[3,2]$, $[2,2]$, $[4,3]$, $[3,3]$ a potom buď $[4,4]$ alebo $[3,4]$. Keby napríklad najskôr nič nekopal a zoskočil rovno na políčko $[4,2]$, nemohol by kopat a zasekol by sa. Všimnite si, že príklady 2 a 3 by sa nemohli vyskytnúť v sadách 1 a 3, pretože nespĺňajú $P = V$.

Vo štvrtom príklade sa na spodok jaskyne dokopať nevieme. Pre zaujímavosť, dokázať to vieme napríklad tak, že v druhom riadku vieme mať voľné najviac tri políčka, v treťom najviac dve a vo štvrtom najviac jedno, ale žiadne z nich nevie byť na ľavom kraji, preto už sa do piateho riadku nedostaneme.

7. Nečakaný výlet

12 b za popis, 8 b za program

Peder sedel pri počítači a nadšene si sťahoval nový diel svojho obľúbeného seriálu Fraiser na svojej obľúbenej stránke uloz-stiahni-to-zadarmo-free-serialy-filmy.tv

Zrazu na neho však vyskočila dáka ruleta a nápis

DOBRY DEN, STE DNEŠNÝM ŠTASTNÝM VÍŤAZOM NAŠEJ ANKETY

Každý týždeň odmeňujeme piatich účastníkov našej ankety cenami hodnoty až **10000 EUR**.

(Peder žiadnu anketu nevyplňal...)

Peder, samozrejme ako každý normálny človek ruletu roztočil, a čo to nevidí? Získava hlavnú cenu: $k + 1$ lístkov do Legolandu! Jeden si nechá určite pre seba. Ale čo s ostatnými?

Úloha

Peder má n kamarátov. i -ty z nich má voľno v dňoch a_i až b_i vrátane. Spočítajte, koľkými spôsobmi vie vybrať k z nich tak, že všetci majú aspoň jeden spoločný voľný deň.

Formát vstupu

V prvom riadku vstupu je číslo n udávajúce počet Pederových kamarátov a k ($1 \leq k \leq n$) – počet lístkov, ktoré má pre kamarátov. Na každom z nasledujúcich n riadkov sa nachádzajú dve celé čísla a_i a b_i – interval voľného času i -teho kamaráta.

V jednotlivých sadách platia nasledujúce obmedzenia:

Sada	1	2	3	4
$1 \leq n \leq$	20	1000	$2 \cdot 10^5$	$2 \cdot 10^5$
$1 \leq a_i \leq b_i \leq$	1 000	10^9	100	10^9

Formát výstupu

Vypíš jeden riadok a v ňom jedno celé číslo – počet spôsobov, ako prideliť kamarátom lístky. Keďže toto číslo môže byť veľmi veľké, vypíšte len jeho zvyšok po delení $10^9 + 7$.

Príklad

vstup

```
3 3
1 4
2 3
2 8
```

výstup

```
1
```

$k = n$, čiže Peder potrebuje zobrať všetkých kamarátov. I keď má dva možné termíny, kedy ich môže zobrať, stále je to tá istá skupina a tak je odpoveď 1.

vstup

```
3 2
1 4
4 9
6 14
```

výstup

```
2
```

Môže zobrať buď prvého a druhého alebo druhého a tretieho.

8. Divné diery

12 b za popis, 8 b za program

Po výbornom farmárskom výkone v minulom kole, sa aj Farmári z Matfyzu vybrali na výlet do Legolandu. Po tom ako sa pár krát stratili v spleti uličiek, jeden zavelil: “Kašlime na to. Spravme z Legolandu strom, nech sa tu nedá zablúdiť.” Rozhodli sa teda, že niektoré cesty v Legolande zastavajú tak aby vznikol strom. Chvíľu rozmýšľali či to spravili optimálne, ale zrazu sa vyskytol oveľa väčší problém.

V Legolande je zopár divných dier v časopriestore, ktoré fungujú takto: Po tom čo kozmickej bytosti zaplatíš c vesmírnych peňazí, ocitneš sa o w sekúnd v mieste, kde sa diera nachádza a to bez ohľadu na to, kde si sa predtým nachádzal. Toto Farmárom otvorilo nové možnosti prepravovania sa v Legolande a teraz ich zaujíma, ako sa vedia za nejaký čas dostať z vchodu do Legolandu napríklad do miestnej lego farmy, alebo na najbližší burger. Keďže však sú študenti, potrebujú to spraviť za čo najmenej vesmírnych peňazí, aby im to ešte fakulta preplatila. Farmári nie sú moc dobrí programátori, preto niet divu, že táto úloha pripadla na vás.

Úloha

Legoland je strom, teda súvislý graf s n vrcholmi a $n - 1$ hranami. Vstup do Legolandu, kde sa nachádzajú Farmári teraz, je vo vrchole 0, v tomto vrchole strom zakoreníme. Pre každú hranu i vieme číslo t_i – čas v sekundách, ktorý trvá prejsť ju Farmárom peši.

V Legolande je $m \leq n$ dier v časopriestore. Diera i sa nachádza vo vrchole $v_i \leq n$ a poznáme pre ňu ešte 2 hodnoty: c_i – cena, ktorú musia Farmári zaplatiť kozmickej bytosti, a w_i – čas v sekundách, ktorý musia Farmári čakať, kým sa premiestnia do vrcholu v_i . Všimnite si, že to nezávisí od toho, kde sa práve Farmári nachádzajú (hovoril som, že diery v časopriestore sú divné?)

Farmári si vybrali q zaujímavých miest, a_1, \dots, a_q a na i -te z nich by sa chceli dostať z vrcholu 0 za b_i času. Vašou úlohou je zistiť, či a za koľko najmenej peňazí sa na miesto a_i vedia dostať v časovom limite b_i .

Formát vstupu

Na prvom riadku vstupu je číslo T popisujúce počet testov. Nasleduje T testov, pred každým z nich je prázdny riadok.

Na prvom riadku testu je číslo n ($1 \leq n \leq 10^5$) – počet vrcholov v strome.

Na druhom riadku je popis stromu: $n - 1$ čísel p_1, \dots, p_{n-1} oddelených medzerou. Číslo p_i ($0 \leq p_i < i$) popisuje, že otec vrcholu s číslom i je vrchol p_i . (Vrcholy číslujeme od 0. Vrchol 0 nemá otca, lebo je koreň.) Je zaručené, že žiaden vrchol nemá za otca vrchol s väčším číslom.

Na treťom riadku sú zadané časy na prejdenie hrán: $n - 1$ čísel t_1, \dots, t_{n-1} oddelených medzerou. Číslo t_i ($1 \leq t_i \leq 10^9$) popisuje čas v sekundách, aký trvá prejdenie z vrcholu i do jeho otca (a naopak).

Na štvrtom riadku je číslo m ($1 \leq m \leq 10^5$) – počet dier.

Nasleduje m riadkov, i -ty z nich obsahuje 3 čísla v_i, c_i, w_i oddelené medzerou ($0 \leq v_i < n$, $1 \leq c_i, w_i \leq 10^9$): postupne vrchol v ktorom sa diera nachádza, cena vo vesmírnych peniazoch a čas v sekundách aký trvá kým sa do diery premiestnite (viac dier môže byť v tom istom vrchole)

Na nasledujúcom riadku je číslo q ($1 \leq q \leq 10^5$) – počet otázok.

Nasleduje q riadkov. Na i -tom z nich sú 2 čísla a_i, b_i ($0 \leq a_i < n$, $1 \leq b_i \leq 10^9$) – postupne vrchol, kam sa chcú Farmári dostať a ako najdlhšie to môže trvať.

Formát výstupu

Pre každú otázku vypíšte jeden riadok a v ňom jedno číslo – koľko najmenej peňazí musia Farmári použiť, na to aby sa dostali z vrcholu 0 do cieľa v časovom limite, prípadne -1 , ak sa tam v časovom limite dostať nevedia.

Hodnotenie

Sú 4 sady vstupov, za každú sú 2 body. Nech C označuje najväčšie z čísel na vstupe a H hĺbku stromu⁴, potom v jednotlivých sadách platia nasledujúce obmedzenia:

Sada	1	2	3	4
$1 \leq n, q \leq$	1000	10^5	10^5	10^5
$1 \leq m \leq$	1000	100	10^5	10^5
$C \leq$	1000	10^9	10^9	10^9
$1 \leq H \leq$	n	n	100	n

Súčet hodnôt n vo všetkých testoch nepresiahne 10^5 . Rovnako aj súčet hodnôt m a q .

Príklady

vstup

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

výstup

```
0
1
```

Pri prvej otázke sa dá v časovom limite prejsť do cieľa peši, preto je odpoveď 0. Pri druhej otázke musíme použiť dieru vo vrchole 1, tá má cenu 1.

vstup

```
2
5
0 1 2 3
4 7 8 6
3
0 7 3
4 2 3
2 1 10
3
3 7
4 14
2 10
5
0 0 2 1
3 1 3 4
2
4 6 2
4 3 3
3
4 2
4 3
4 7
```

výstup

```
-1
2
1
6
3
0
```

⁴Hĺbku stromu definujeme ako maximálny počet hrán, po ktorých vieme prejsť na ceste z koreňa, pokiaľ nemôžeme po hrane prejsť viac krát a nepoužijeme diery.