

Korešpondenčný seminár z programovania

Leták zimnej časti XLIV. 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 zdokonaliť 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.
- Ťeš 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 mesiac a pol 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 neprezrádzaj riešenia ostatným riešiteľom. Odpisovanie riešení a prezradenie riešení pred termínom kola je porušením pravidiel KSP. Po skončení kola sa, samozrejme, o riešeniach rozprávať môžeš. :)
- **Odovdzaj** 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 vyzerať 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 pamäťovej zložitosti programu. Viac sa dozvieš na stránke `https://www.ksp.sk/ako-riesit`. Popis opravia 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).

Účasťou 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.

Korešpondenčný seminár z programovania
XLIV. ročník, 2026/27
FMFI UK,
Mlynská Dolina, 842 48 Bratislava

Úlohy 2. kola zimnej časti

Termín odoslania riešení tohto kola je **14. decembra 2026**. Doprogramovávanie končí 28. decembra 2026.

1. Presunuté povinnosti

12 b za popis, 8 b za program

KSP vedúci v decembri zvyknú rozkladať a zdobiť stromček v T2[1]. Tento rok sme však nestíhali vydávať kolá a kupovať počítače, a aby sme stihli ďalšie kolo pripraviť včas, rozhodli sme sa začať už teraz.

Teraz ale nestíhame zdobiť stromček, preto túto úlohu prenecháme vám. Dostanete veľkosť stromčeka a tradičné ozdoby[2] zavesené na dlhej šnúre, vašou úlohou je šnúru rovnomerne zavesiť okolo stromčeka.

Ako motiváciu vám posielame minuloročnú fotku:



Úloha

Na vstupe dostanete dĺžku ozdoby n a výšku stromčeka v. Vašou úlohou bude zavesiť ozdobu na stromček. Ozdobu treba zavesiť na každý nepárny riadok odspodu.

Je zaručené, že ozdoba bude mať správnu dĺžku a na stromček bude presne sedieť.

Formát vstupu

V prvom riadku vstupu je číslo n a v, ktoré udávajú dĺžku ozdoby a výšku stromčeka.

Na ďalšom riadku je n znakov popisujúcu ozdobu.

V jednotlivých sadách platia nasledujúce obmedzenia:

[1] Sídlo KSP na Matfyze.
[2] Všetko od diskiet, cez ethernetové káble, až po myšky a počítače.

Sada	1	2
$6 \leq n \leq$	55	500 500
$3 \leq v \leq$	10	1 000

Formát výstupu

Vypíšte $v + 1$ riadkov popisujúcich vami ozdobený stromček.

Dávajte si pozor na správne medzery na začiatku riadkov a nepíšte medzery na konce riadkov.

Príklad

vstup

```
15 5
-0-0--0--0-0-0X
```

výstup

```
    X
   ***
  0-0-0
 *******
-0-0--0--
    #
```

vstup

```
21 6
*CHRISTMAS**MERRY*KSP
```

výstup

```
     *
    KSP
   *****
  *MERRY*
 *********
*CHRISTMAS*
     #
```

2. Odkiaľ vyraziť

12 b za popis, 8 b za program

Nové počítače sú drahé. Našťastie môžeme *Kupovať Staré Počítače*, a tak ušetriť veľa peňazí. Tie sa vždy zídu na ďalšie počítače. Ralbo listuje stránkami svojho obľúbeného bazárového portálu a značí si výhodné ponuky.

Dotykový displej, dotyk nefunguje… cena 30€… ako zadarmo, displej opravím aj keby nechcem… *intel i5, ôsma generácia, windows 11…* úbohé stvorenie, to je snáď týranie starých procesorov…

Po hodinách strávených vyberaním kandidátov na vzkriesenie najlepším operačným systémom sa Ralbo konečne dostáva k plánovaniu cesty. Už už hádže adresy do máp, keď si na stránke všimne neznáme tlačítko.

Bazár má novú experimentálnu funkciu. Tak toto je isto navibekódené, pomyslí si. Pre zadané inzeráty vám poskyne palivo, ktoré presne vystačí na navštívenie všetkých miest a návrat do prvého. Má to ale malý háčik, nie z každého mesta vystačí benzín v nádrži do ďalšieho. Kupujúci si musí rozumne vybrať, kde chce svoju cestu začať. Taktiež vaše auto má trošku väčšiu spotrebu ako zvyčajné autá.

Úloha

Dostanete zoznam staníc, ktoré sú v jednom kruhu. Pre každú stanicu číslo c a d, kapacitu danej stanice - koľko litrov v nej vieme natankovať paliva a vzdialenosť od nasledujúcej stanice v km. Zoznam staníc tvorí uzavretý okruh (z n-tej stanice sa ide opäť do 1. stanice). Naše auto má spotrebu 1 l/km. Súčet palív a vzdialeností je rovnaký. Vašou úlohou je určiť, v ktorej stanici musíme začať, aby sme okruh prešli celý. Začíname s prázdnou nádržou. Kapacita nádrže nie je obmedzená.

Formát vstupu

V prvom riadku vstupu je číslo n ($1 \leq n \leq 10^9$) udávajúce počet staníc.

Nasledujúcich n riadkov obsahuje dvojicu čísel c a d oddelených medzerou.

Sada	1	2
$1 \leq n \leq$	10^3	10^8
$1 \leq s \leq$	10^9	10^{12}

Kde s je súčet paliva = súčet vzdialeností.
Poznámka: Pri programovaní v C++ sa súčet paliva v druhej sade nemusí zmestiť do 32-bitového intu.

Formát výstupu

Vypíšte jedno číslo, udávajúce poradové číslo stanice, v ktorej musíme začať. Stanice sú číslované od 1 v poradí zo vstupu.

Príklad

vstup	výstup
4 3 5 15 1 2 1 7 20	2

Ak začneme na druhej stanici, natankujeme si 15 litrov benzínu. To nám vystačí až po 4. stanicu, pričom po ceste môžeme postupne dotankovávať palivo, vďaka čomu vieme prejsť až do 1. stanice. Keď tam natankujeme zvyšné 3 litre paliva, v nádrži nám zostane presne 5 litrov, čo nám stačí na návrat do 2. stanice, kde sme začínali.

vstup	výstup
6 1 10 11 20 30 5 2 6 5 5 1 4	3

3. Čet s rodinou

12 b za popis, 8 b za program

Autor úlohy filip.sivicek@trojsten.sk

Ekonomická situácia je zlá - cena ramky sa každým dňom zvyšuje a zvyšuje. Andrejík, ten malý fiškus sa rozhodol, že toto využije pre svoj prospech. Obehá všetkých svojich rodinných príslušníkov, vyprosíka si od nich ramku a predá ju. Jeho rodinný príslušníci bohužiaľ spolu komunikujú. Keď ich navštívi príliš veľa, ostatní mu už nebudú ochotný predať ďalšie.

Našiel výbornú stránku, kde ich predá - Kvadratické Súčty Produktov. Na KSP rozpoznávajú 26 typov ramky. Sú označené písmenami veľkej anglickej abecedy Tam je cena jednej ramky rovnaká ako počet, ktorý im predal. Koľko najviac vie Andrejík zarobiť?

Úloha

Na vstupe dostanete počet rodinných príslušníkov a koľko mu dá ramku pred tým, než ju začnú predávať oni sami. Okrem toho dostanete reťazec n znakov, v ktorom i-ty znak predstavuje typ ramky i-teho počítača.

Zistite, koľko najviac môže Andrejík za tento deň zarobiť.

Vstup

Prvý riadok vstupu obsahuje prirodzené čísla n a k ($1 \leq k \leq n \leq 1\,000\,000$).

Druhý riadok obsahuje n veľkých písmen anglickej abecedy – typy ramiek, ktoré rodinný príslušníci.

Výstup

Vypíšte jeden riadok a v ňom jedno číslo – najväčší súčet cien, ktorý vie Andrejík zarobiť.

Príklady

vstup	výstup
15 10 DZFDFZDFDDDDDDF	82

vstup	výstup
6 4 YJSNPI	4

V prvom príklade prinesie 9 krát `D` a raz `Z`. V druhom prípade prinesie ramky 4 rôznych typov.

4. Ísť po počítač

12 b za popis, 8 b za program

Mladý Emanuel má jasný životný cieľ – kúpiť si nový, neuveriteľne výkonný herný počítač, aby sa mohol hrať s Jozefom online hry. To však má jeden háčik - nie je medzi nimi natiahnutá sieť, teda nevedia spolu rozumne rýchlo komunikovať. Musí sa teda vybudovať nová optická sieť medzi nimi.

Sieťari vybudujú a otvoria stanicu i, až keď do nej vedie aspoň r_i plne dokončených optických liniek, skôr sa im to neoplatí. Otvorenie stanice zároveň naštartuje ďalších, ktorí sa hneď pustia do budovania všetkých zvyšných plánovaných liniek vedúcich z nej. Stavba každej linky trvá rok, takže sa spojazdní až od nasledujúceho roka. Zopár nadšených staníc je otvorených odjakživa a všetky linky z nich sú už dávno hotové.

Pomôžte Emanuelovi zistiť, za koľko rokov sa najskôr dokáže spojiť s Jozefom a hrať sa svoju obľúbenú hru - kedy si má ten herný počítač kúpiť.

Úloha

Máme danú krajinu s n stanicami očíslovanými od 0 po $n-1$ a m obojsmernými linkami medzi nimi. Každá stanica i má stanovený limit otvorenia $r_i \geq 0$.

- Každá stanica s limitom $r_i = 0$ je otvorená odjakživa a všetky linky vedúce z nej sú už dokončené, prevádzkyschopné od roku 0.
- Každá ostatná stanica sa otvorí v najskoršom roku, v ktorom je aspoň r_i liniek vedúcich z nej dokončených.

Keď sa takáto stanica otvorí v roku y, začne stavať všetky svoje linky, ktoré doteraz neboli dokončené, a tie sú prevádzkyschopné od roku $y+1$. Linku medzi dvoma stanicami teda postaví tá z nich, ktorá sa otvorí skôr. Stanica, ku ktorej nikdy nepovedie aspoň r_i dokončených liniek, sa neotvorí nikdy.

Emanuel je napojený k stanici s a Jozef býva pri stanici t. Dáta môžu tiecť iba cez už postavené linky a už otvorené stanice. Zistite, za koľko najmenej rokov bude Emanuel vedieť komunikovať s Jozefom cez novú optickú sieť.

Formát vstupu

V prvom riadku vstupu sú štyri celé čísla n, m, s, t ($1 \leq n \leq 200\,000$, $0 \leq m \leq 200\,000$, $0 \leq s, t \leq n-1$), kde n je počet staníc, m počet liniek, s Emanuelova stanica a t Jozefova stanica. V druhom riadku je n celých čísel $r_0, r_1, \ldots, r_{n-1}$ ($0 \leq r_i \leq n$), kde r_i udáva limit otvorenia stanice i. Nasleduje m riadkov, z ktorých každý obsahuje dve celé čísla u a v ($0 \leq u, v \leq n-1, u \neq v$), čo znamená, že medzi stanicami u a v je naplánovaná obojsmerná linka. Medzi žiadnymi dvoma stanicami nevedie viac ako jedna linka a žiadna linka nevedie zo stanice do nej samej.

Formát výstupu

Vypíšte jedno celé číslo – za koľko najmenej rokov Emanuel vie začať komunikovať s Jozefom, teda vie poslať správu zo stanice s do stanice t. Ak sa to nestane nikdy, vypíšte -1.

Hodnotenie

V jednotlivých testovacích sadách platia nasledujúce obmedzenia:

Sada	1	2	3	4	5	6	7	8
$1 \leq n, m \leq$	50	1 000	200 000	200 000	200 000	200 000	100 000	200 000

Navyše v 3. sade platí, že všetky $r_i \leq 1$, vo 4. sade tvoria trate súvislý strom, teda $m = n-1$, v 5. sade platí, že súčet všetkých r_i nepresiahne 2 000, a v 6. sade sa zaručene každá stanica raz otvorí a odpoveď nie je -1.

Príklady

vstup

```
5 6 0 4
0 1 2 1 2
0 1
0 3
1 2
3 2
2 4
1 4
```

výstup

```
2
```

Stanica 0 v roku 0 sprevádzkuje linky do staníc 1 a 3, čím obe okamžite otvorí. V roku 1 stanice 1 a 3 dokončia svoje linky do stanice 2 a otvoria ju, vďaka čomu sa v roku 2 dokončí aj druhá linka do cieľovej stanice 4 a Emanuel do vie začať komunikovať s Jozefom po trase $0 \to 1 \to 2 \to 4$.

vstup

```
5 4 0 4
0 1 3 0 1
0 1
1 2
3 4
2 4
```

výstup

```
-1
```

Stanica 0 v roku 0 otvorí stanicu 1 a stanica 3 rovnako otvorí stanicu 4. Spojovacia stanica 2 však vyžaduje až 3 dokončené trate, no vedú do nej iba 2 trate (zo staníc 1 a 4), takže sa nikdy neotvorí a Emanuel sa z Jozefom nikdy nespojí.

5. Tajomná jaskyňa

12 b za popis, 8 b za program

Miškin počítač už má svoje roky a začalo jej vadiť, že je strašne pomalý. Preto sa rozhodla zaobstarať si nový a po chvíli[3] vyberania si konečne jeden objednala. Hneď ako jej nový počítač doručili, povedala si, že ho rovno otestuje pri hraní svojej obľúbenej hry.

V hre sa Miška nachádza v jaskyni plnej netopierikov. Dajú sa v nej nájsť aj rôzne predmety, vzácne aj menej vzácne, napríklad netopierie mlsky. Sem-tam sa k nej priblíži roztomilý netopierik natoľko, že ho môže nakŕmiť takouto mlskou. Ak to spraví, tak netopierika automaticky ochočí a už navždy bude jej verným domácim maznáčikom. Ak ho však nenakŕmi, netopierik veľmi rýchlo odletí a Miška už nikdy nedostane príležitosť ochočiť si ho.

Miška má veľmi rada netopierikov a chce maximalizovať súčet roztomilostí tých, ktorých ochočí. Rada by preto vedela, aký najväčší súčet roztomilostí mohla v tejto hre získať. Pomôžte jej s tým.

Úloha

Na začiatku hry Miška nemá žiadne mlsky. Počas hry sa v jaskyni postupne udeje n udalostí dvoch typov:

- 0 – Miška nájde x_i mlsiek.
- 1 – Netopierik s roztomilosťou k_i sa priblíži k Miške dostatočne blízko na to, aby ho mohla ochočiť za jednu mlsku. Ak ho neochočí, odletí ešte predtým, než nastane ďalšia udalosť.

Zistite, aký najväčší súčet roztomilostí ochočených netopierikov mohla Miška dosiahnuť pri danom priebehu udalostí. Nezabudnite, že Miška dokáže ochočiť netopierika iba vtedy, keď má v čase jeho priblíženia aspoň jednu mlsku, ktorú ochočením spotrebuje.

Formát vstupu

V prvom riadku vstupu je počet udalostí n ($1 \leq n \leq 2 \times 10^5$).

Nasleduje n riadkov opisujúcich jednotlivé udalosti, ktoré číslujeme od 0 po $n-1$. Každý z týchto riadkov obsahuje dve celé čísla. Prvé číslo t_i označuje typ i-tej udalosti:

- Ak $t_i = 0$, druhé číslo x_i ($0 \leq x_i \leq 10^6$) predstavuje počet mlsiek, ktoré Miška práve našla.
- Ak $t_i = 1$, druhé číslo k_i ($0 \leq k_i \leq 10^9$) predstavuje roztomilosť netopierika, ktorého vtedy stretla.

[3] Bolo to troošku viac než chvíľa, ale to budeme zanedbávať.

V jednotlivých sadách platia nasledujúce obmedzenia:

Sada	$1 \leq n \leq$	Dodatočné obmedzenia
1	20	Roztomilosti sú najviac 10^6 ($k_i \leq 10^6$)
2	20	-
3	1 000	Miška nájde všetky mlsky práve v prvej udalosti ($t_i = 0 \iff i = 0$)
4	1 000	Roztomilosti sú nerastúce ($i < j \implies k_i \geq k_j$)
5	1 000	-
6	2×10^5	Miška nájde všetky mlsky práve v prvej udalosti ($t_i = 0 \iff i = 0$)
7	2×10^5	Roztomilosti sú nerastúce ($i < j \implies k_i \geq k_j$)
8	2×10^5	-

Formát výstupu

Vypíšte jeden riadok a v ňom jedno celé číslo: najväčší súčet roztomilostí ochočených netopierikov, ktorý dokáže Miška dosiahnuť.

Upozornenie: Výsledok ani niektoré iné premenné sa nemusia zmestiť do 32-bitovej premennej.

Príklady

vstup

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

výstup

```
4
```

Miška počas celej hry našla iba jednu mlsku, preto mohla ochočiť najviac jedného netopierika. Samozrejme si chcela vybrať takého, ktorý mal čo najväčšiu roztomilosť. Netopierika s roztomilosťou 7 však stretla ešte pred nájdením mlsky, preto bol najlepšou možnosťou netopierik s roztomilosťou 4.

vstup

```
3
0 3
1 5
1 5
```

výstup

```
10
```

Miška hneď na začiatku našla až tri mlsky, ale stretla iba dvoch netopierikov. Mohla teda ochočiť oboch a dosiahnuť súčet roztomilostí 10.

6. A tento nechceš?

12 b za popis, 8 b za program

Biznismeni z konkurenčnej organizácie Kraksne Staré Predávame našli spôsob, ako dosiahnuť, aby si od nich každý kúpil dokonalý notebook. Ich tajomstvo je nasledovné: žiadny dokonalý notebook neexistuje.

Inak povedané, každý majiteľ notebooku, ktorý svoje zariadenie používa viac než týždeň, si je vedomý všetkých jeho nedostatkov… a so správnou porciou manipulácie sa nechá presvedčiť o tom, aby ho vymenil za iný model, ktorý žiadne z nich nemá. Nie že by ten nový notebook nemal kopu ďalších problémov – ale v tom práve spočíva tá dokonalá biznisová stratégia! Keď dokážeme presvedčiť každého vlastníka notebooku, že je ochotný vymeniť ho za akýkoľvek iný, … výsledok? Veľa peňazí!

Teda, takto by to fungovalo v dokonalom svete. V realite nie je možné presvedčiť ľudí o každej výmene modelu za model. KSP má však času dosť, takže na dosiahnutie svojich cieľov je ochotné poradiť každému človeku ľubovoľne mnoho výmen. A pokiaľ to nebude stačiť, ešte stále je tu možnosť použiť nejaký "špinavý trik" (na rozidel od všetkého spomenutého doteraz), napríklad písať fake reviews na stránky online obchodov alebo spamovať diskusné fóra, aby si ľudia mysleli, že nejaký notebook má alebo nemá kľúčové nedostatky. To je však náročné, drahé a podozrivé, preto chce KSP využívať tieto spôsoby čo najmenej.

Úloha

KSP o každom modeli notebooku vie, za ktoré iné modely sú ho jeho majitelia ochotní vymeniť. Jeho cieľom je, aby dokázalo presvedčiť majiteľa akéhokoľvek notebooku na výmenu s akýmkoľvek iným notebookom. Na to, aby dosiahlo požadovaný výsledok, môže pritom spraviť ľubovoľný počet výmen.

KSP môže využívať špinavé triky. Dôsledkom jedného triku je, že majitelia konkrétneho modelu notebooku budú ochotní vymeniť ho za iný konkrétny model. KSP chce však využiť špinavých trikov čo najmenej.

Zistite, aký najmenší počet špinavých trikov musí KSP využiť, aby bolo možné sériou výmen presvedčiť majiteľa ktoréhokoľvek modelu notebooku, nech ho vymení za akýkoľvek iný model. Teda, aby pre každý cieľový model, a ľubovoľný začiatočný model existovala postupnosť výmen, po ktorých zákazník skončí s cieľovým modelom.

Formát vstupu

Na prvom riadku vstupu sú dve čísla oddelené medzerou – $1 \leq n \leq 10^5$, počet rôznych modelov notebookov. Nasleduje n riadkov s číslami oddelenými medzerami. Riadok i začína počtom modelov notebookov, za ktoré sú vlastníci modelu i ochotní svoj notebook vymeniť. Za ním je zoznam týchto modelov. Modely sú číslované postupne od 1. Súčet počtov modelov v jednotlivých riadkoch nepresiahne 10^6.

Úloha má štyri testovacie sady. V prvých troch sadách platia tieto dodatočné obmedzenia:

- V prvej sade platí, že pokiaľ je možné majiteľa jedného modelu notebooku presvedčiť o výmene za iný model, je vždy možné rovno ho presvedčiť aj o opačnej výmene.
- V druhej sade platí, že pokiaľ je možné majiteľa jedného notebooku presvedčiť o výmene za iný model, tak ho žiadnou postupnosťou výmen nie je možné presvedčiť o opačnej výmene (to je, samozrejme, iba počiatočný stav, úloha zostáva rovnaká).
- V tretej sade platí, že počet modelov notebookov n je najviac 1000.

Formát výstupu

Vypíšte jeden riadok a v ňom jedno celé číslo – najmenší počet špinavých trikov, ktoré musí KSP spraviť, aby dokázalo kohokoľvek presvedčiť o výmene notebooku za akýkoľvek iný model.

Príklady

vstup

```
3
2 2 3
0
1 2
```

výstup

```
1
```

Majitelia prvého modelu sú ochotní vymeniť ho za ľubovoľný zo zvyšných dvoch. Tí, čo majú tretí model, sú ochotní vymeniť ho za druhý, ale nie za prvý, a tí, čo majú druhý, ho nie sú ochotní vymeniť vôbec. Stačí však jedným špinavým trikom presvedčiť majiteľov druhého modelu notebooku, že sa oplatí vymeniť ho za prvý model, a bude možné presvedčiť kohokoľvek na akúkoľvek výmenu (napríklad majiteľov tretieho modelu sa dá presvedčiť o výmene s prvým modelom tak, že sa najprv presvedčia o výmene s druhým modelom, a ten potom vymenia za prvý).

vstup

```
3
1 2
1 3
1 1
```

výstup

```
0
```

vstup

```
5
1 3
1 3
2 4 5
0
0
```

výstup

```
2
```

7. Čo si dnes kúpi Dalibor

12 b za popis, 8 b za program

Dalibor, tiež nazývaný Dalyb, je veľký swagger. Každých pár dní musí mať novú designerskú tašku, aby bol jeho outfit dostatočne fly. Akurát aj Dalyb žije v tejto ekonomike a veľkú časť príjmov musí dať na obyčajné smrteľnícke výdavky ako nájom a potraviny. Preto si tašky radšej len požičiava.

Aj tak ale vreckové od jeho rodičov nestačí pre potreby jeho swagu. Rozhodol sa teda hustliť predávaním operačnej pamäte. V RAMSWAGSHOPe vedia oceniť jeho štýl. Každý deň, keď prvýkrát príde s nejakou jeho designerskou taškou, ponúknu mu zadarmo RAMku určitého objemu a určitej hodnoty. Pokiaľ sa mu táto RAMka do tašky zmestí, môže si ju zobrať.

Dalyb už má vyhliadnuté nejaké designerské tašky, ktoré by si chcel požičať. Pre každú vie, kedy si ju plánuje požičať, kedy vrátiť a aký má objem. Pri vrátení dostane toľko peňazí, akú budú mať hodnotu RAMky uložené vo vracanej taške. Samozrejme, musia sa do tašky zmestiť, teda súčet objemov RAMiek v taške nemôže prevyšovať objem tejto tašky. Pomôžte Dalybovi pre každú tašku zistiť, koľko najviac je ňou schopný zarobiť.

Úloha

Máme daných n predmetov, kde každý predmet má svoj objem a hodnotu. Ďalej dostaneme q otázok. i-ta z nich určená číslami l_i, r_i, c_i. Jej odpoveď je jedno číslo: najväčšia možná hodnota predmetov, ktoré vieme zobrať, ak nemôžu mať objem väčší ako c_i a môžeme brať iba z predmetov, ktoré boli na vstupe uvedené od l_i-teho až po r_i-ty predmet vrátane.

Formát vstupu

V prvom riadku vstupu je číslo n udávajúce počet predmetov na vstupe. Nasleduje n riadkov. Na každom sú dve čísla v_i, p_i oddelené medzerou, ktoré reprezentujú objem a hodnotu i-teho predmetu. Platí, že $1 \leq v_i, p_i \leq 10^9$.

Na ďalšom riadku je číslo q udávajúce počet otázok. Nasleduje q riadkov. Na každom sú tri čísla l_i, r_i, c_i, ktorých význam je popísaný vyššie. Platí, že $1 \leq l_i \leq r_i \leq n$ a $1 \leq c_i \leq 500$.

Vstupy sú rozdelené do ôsmich sád. Za každý vyriešenú sadu získa váš program jeden bod. V jednotlivých sadách platia nasledujúce obmedzenia:

Sada	1, 2	3, 4	5, 6	7, 8
$1 \leq n \leq$	100	2 000	2 000	$2 \cdot 10^4$
$1 \leq q \leq$	100	$5 \cdot 10^4$	$5 \cdot 10^4$	$2 \cdot 10^5$

V druhej sade navyše platí, že pre všetky $1 \leq i \leq n$ platí $r_i - l_i \geq \lfloor N/2 \rfloor$.

Formát výstupu

Vypíšte q riadkov. V i-tom z nich bude odpoveď na i-tu otázku zo vstupu

Príklady

vstup

```
4
5 5
2 4
4 3
3 2
3
1 4 10
1 4 9
2 4 1
```

výstup

```
11
9
0
```

Tento vstup sa mohol vyskytnúť aj v druhej sade, lebo všetky otázky sa pýtaju na dostatočne veľký podúsek. V prvej otázke je optimálne zobrať prvý, druhý a štvrtý predmet v poradí. V je optimálne zobrať prvý a druhý predmet. V tretej otázke nevieme zobrať žiadny predmet, keďže majú všetky väčší objem ako môžeme zobrať. Odpoveď je teda 0

vstup

```
8
145 512438721
31 784215630
219 365190442
87 913726504
42 658341297
268 194827365
176 845613209
24 731905418
8
5 8 150
2 6 100
1 7 300
3 8 250
4 5 130
1 4 220
6 8 400
2 2 50
```

výstup

```
1390246715
1442556927
2543555343
2303973219
1572067801
1697942134
1577518627
784215630
```

Chýba súbor zad-prikl8.tex!!!