



Úlohy 2. kola zimnej časti

Termín odoslania riešení tohto kola je pondelok 11. Februára 2019. Doprogramovanie končí 25. Februára 2019.

1. Terms and conditions

12 b za popis, 8 b za program

Ešte pred pár mesiacmi všetky firmy menili svoje pravidlá narábania s osobnými údajmi. Preto všetci chceli, aby obyčajní používatelia ako Denis tieto nové pravidlá odsúhlasili. V tom čase to Denis vôbec neriešil. Teraz sa však rozhodol, že už nezaklikne ani jedno políčko. Veď tým nebude zbytočne strácať čas! A preto vymyslel geniálny ťah. Vytvorí program, čo to zaňho odklikne. No uvedomil si dôležitý fakt. Keď to dá ako úlohu do KSP nemusí to ani len programovať!

Úloha

Napište Denisovi program, ktorý v texte nájde políčko, ktoré vyzerá presne takto:

```
+++\n| |\n+++
```

a zakreslite doň veľké písmeno X.

Formát vstupu

Na začiatku vstupu sa nachádza číslo $n - 3 \leq n \leq 1000$ – počet riadkov na vstupe. Nasleduje n riadkov textu. Na každom riadku sa nachádza m znakov – $1 \leq m \leq 100$. Medzi nimi sa bude nachádzať práve jedno políčko na zaškrtnutie. Znak, ktoré sa v texte budú nachádzať, budú znaky z [ASCII tabuľky](#)¹. Konkrétne, pre každý znak c platí, že je z rozsahu $32 \leq c \leq 126$ alebo je znakom nového riadku (10).

Formát výstupu

Na výstup vypíšete rovnaký text, aký ste dostali na vstupe (bez prvého riadku, v ktorom sa nachádza n), s tým rozdielom, že políčko bude zaškrtnuté.

Príklady

vstup

```
15\nV zmysle zakona Slovenskej republiky\nc. 18/2018 Z. z. o ochrane osobnych\naudajov a o zmene a doplneni niekterych\nezakonov, ktory najma v §19-30 upravuje\nprava dotknutej osoby v oblasti\nspracovania osobnych udajov, vyjadrujem\nsuhlas so spracovanim poskytnutych osobnych\nudajov.\n+++\n| |\n+++\nUvedeny suhlas sa tyka aj poskytnutia\nuvedenych udajov tretim stranam za ucelom\nmierenych reklam na iste druhy pracich\nprostriedkov.
```

výstup

```
V zmysle zakona Slovenskej republiky\nc. 18/2018 Z. z. o ochrane osobnych\naudajov a o zmene a doplneni niekterych\nezakonov, ktory najma v §19-30 upravuje\nprava dotknutej osoby v oblasti\nspracovania osobnych udajov, vyjadrujem\nsuhlas so spracovanim poskytnutych osobnych\nudajov.\n+++\n|X|\n+++\nUvedeny suhlas sa tyka aj poskytnutia\nuvedenych udajov tretim stranam za ucelom\nmierenych reklam na iste druhy pracich\nprostriedkov.
```

¹<https://sk.wikipedia.org/wiki/ASCII>

vstup

```
13
Tymto potvrdzujem, ze nie som robot
ani program na zaklikavanie policok
  +-+  +_+
  | |  | |
  +_+  +-+

a navyse suhlasim so vsetkymi podmienkami
ktore som si precital v prilozenych
47 stranach legalneho balastu

click  +-+ here
to     | | accept
terms  +-+ & conditions
```

výstup

```
Tymto potvrdzujem, ze nie som robot
ani program na zaklikavanie policok
  +-+  +_+
  | |  | |
  +_+  +-+

a navyse suhlasim so vsetkymi podmienkami
ktore som si precital v prilozenych
47 stranach legalneho balastu

click  +-+ here
to     |X| accept
terms  +-+ & conditions
```

2. Iregulárna strava

12 b za popis, 8 b za program

Janko sa na internáte delí o chladničku s ďalšími tromi ľuďmi. Často sa mu stáva, že jedlo, ktoré si do nej odloží, mu odtiaľ záhadným spôsobom zmizne. Napríklad, v minulú nedeľu si z domu doniesol lahodné grilované kuracie stehno s ryžou a nivovou omáčkou. Odložil ho na svoju policičku v chladničke s tým, že ho zje v pondelok večer.

V pondelok večer o 19.00 sa po celom dni na fakulte, hladný ako vlk, vrátil na internát. Otvoril chladničku a čo tu vidí? Nič! Stehno a jeho predstavy o lahodnej večeri sú preč.

Povedal si, že to už takto ďalej nemôže pokračovať, a prišiel s nápadom: Uloží si do chladničky okrem jedlých vecí aj veci nejedlé. Všetky veci zabalí do alobalu tak, aby spolubývajúci nevedeli, čo je dnu.

Problém je, že takto ani on nebude vedieť, čo môže bezpečne zjesť. Našťastie, Janko sa nedávno v škole učil o šifrách a tak pozná spôsob, ako si označí balíčky tak, aby ich obsah poznal iba on.

Vytvoril si teda tabuľku, v ktorej každému malému písmenu anglickej abecedy priradil práve jedno veľké písmeno anglickej abecedy. Balíčky si označil dvoma slovami. Prvé sa skladá z malých písmen anglickej abecedy a druhé z veľkých písmen anglickej abecedy. Ak je v balíku jedlo, tak druhé slovo je vytvorené z prvého podľa tejto tabuľky.

Jankovi spolubývajúci, ktorí nepoznajú túto šifru, si tak dosť pravdepodobne pochutia na nejedlých balíčkoch, ktoré môžu obsahovať napríklad drevo alebo kapsuly s pracím práškom.

Napíšte Jankovi program, ktorý mu pomôže zistiť, či je daný balíček jedlý.

Úloha

Na vstupe je zoznam balíčkov v chladničke. Na každom z nich sa nachádzajú práve dve slová. Zistite podľa slov na každom balíčku, či je v ňom jedlo. V balíčku sa nachádza jedlo práve vtedy, keď platí:

- Každému písmenu v prvom slove je pridelené práve jedno veľké písmeno (obraz) v druhom slove.
- Rovnaké písmená majú rovnaký obraz.
- Rozdielne písmená majú rozdielny obraz.
- Poradie obrazov v druhom slove zodpovedá poradiu písmen v prvom slove.

Formát vstupu

Na prvom riadku vstupu sa nachádza číslo $1 \leq t \leq 10^4$, počet balíčkov v chladničke. Nasleduje t popisov balíčkov – dva riadky obsahujúce slová na jednotlivom balíčku. Prvé slovo je tvorené malými, a druhé veľkými písmenami anglickej abecedy. Každé slovo obsahuje aspoň jeden znak. Platí, že súčet dĺžok všetkých slov nepresahuje 4 000 000.

Formát výstupu

Pre každý balíček na výstup vypíšte “ano”, ak je v ňom jedlo, inak vypíšte “nie”.

Príklady

vstup

```
4
anna
ABBA
anna
ABB
lopta
HOKEJ
banany
ANANAS
```

výstup

```
ano
nie
ano
nie
```

Zo slova "anna" sa 'a' zobrazilo do 'A' a 'n' do 'B'

Slovo "ABB" je kratšie než "anna", takže neplatí, že každé písmeno z prvého slova sa **práve raz** zobrazí do druhého.

V slove "lopta" sa žiadne písmeno neopakuje, takže päť písmen sa zobrazilo do piatich rôznych obrazov.

Slovo "banany" sa nezobrazilo správne do slova "ANANAS", keďže až dvom písmenám 'b' a 'n' je priradené 'A'.

3. Dohrajte hru

12 b za popis, 8 b za program

Baklažán a Buj sa hrajú hru. Zoberú balíček kariet a rozdelia ho na kôpky. Každá karta má na sebe napísané číslo predstavujúce body. Potom striedavo ťahajú z kôpok, pričom hráč na ťahu si vždy zvolí z ktorej kôpky odoberie kartu. Aby to nebolo príliš zložité, tak sa dohodli, že Baklažán bude brať kartu iba z vrchu a Buj iba zo spodu. Hra sa končí vtedy, keď sa minú karty a vyhráva ten s najväčším súčtom bodov.

Obaja majú ale veľa povinnosti v škole. Chceli by vás teda poprosiť aby ste zopár hier odohrali za nich. Odkazujú, že by obaja hrali optimálne.

Úloha

Zistite koľko bodov bude mať Baklažán a Buj po skončení hry ak obaja budú hrať optimálne. Hru vždy začína Baklažán.

Formát vstupu

Na prvom riadku vstupu sa nachádza číslo n ($1 \leq n \leq 100$), počet kôpok kariet. Za ním nasleduje n riadkov. V každom nasledujúcom riadku je číslo s_i ($1 \leq s_i \leq 100$) označujúce počet kariet v i -tej kôpke. Za ním nasleduje s_i čísel c_1, c_2, \dots, c_{s_i} , hodnoty kariet od vrchu kôpky až po spodok. Pre ľubovoľné c_k platí, ($1 \leq c_k \leq 1000$).

Čiastočné body môžete získať v sade 1, kde je vždy počet kôpok rovný 1. Pre sadu 2 platí, že počet kôpok je 2.

Formát výstupu

Na výstup vypíšete dve čísla. Počet bodov Baklažána a počet bodov Buja po ukončení hry.

Príklady

vstup

```
2
1 100
2 1 10
```

výstup

```
101 10
```

Baklažán si zobral kartu 100, Buj 10 a nakoniec Baklažán 1.

vstup

```
1
9 2 8 6 5 9 4 7 1 3
```

výstup

```
30 15
```

Baklažán by si zobral karty 2, 8, 6, 5, 9 a Buj 3, 1, 7, 4.

vstup

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

výstup

18 18

Hra by mohla prebiehať napríklad takto:

Baklažán zoberie kartu z tretej kôpky (skóre 5:0)

Bui zoberie kartu z druhej kôpky (skóre 5:7)

Baklažán zoberie kartu z druhej kôpky (skóre 13:7)

Bui zoberie kartu z tretej kôpky (skóre 13:13)

Baklažán zoberie kartu z tretej kôpky (skóre 17:15)

Bui zoberie kartu z prvej kôpky (skóre 17:15)

Baklažán zoberie kartu z prvej kôpky (skóre 18:15)

Bui zoberie kartu z prvej kôpky (skóre 18:18)

4. Energetické pole

12 b za popis, 8 b za program

Paulínka si ako každý čerstvý vysokoškolák dopraje dostatočné a zdravé množstvo spánku. Navyše, v záchvate mánie (nebojte sa, mánia už dávno skončila) začala ješť na večeru divné veci. Kombináciou týchto javov sa jej uprostred noci sníval prečudesný sen. Ocitla sa na ostatnom Letnom Tábore Trojstenu (na tom o Power Rangeroch), kde sa práve na poli hrala hra – strategická behačka.

Družinky účastníkov bojovali o ovládnutie políčok veľkého energetického poľa, ktoré bolo rozdelené na štvorčeky. Prebiehalo to tak, že družinky vysielali svojich rangerov na behy skrz štvorcovú mriežku. Títo rangeri mohli behať iba pozdĺž osí mriežky a nemohli meniť smer. Navyše, aby sa náhodou nezrazili, mohol bežať iba jeden ranger naraz.

Rangeri boli, samozrejme, farební. Keď ranger prebehol cez políčko, políčko sa sfarbilo jeho farbou, a tak bolo ihneď viditeľné, že políčko začalo patriť jeho družinke. Políčka na začiatku hry nemali žiadnu farbu. Aj keď ranger prebehol po už zafarbenom políčku, toto políčko po prebehnutí farbu zmenilo.

Paulínka na túto hru prišla neskoro a celkom by ju zaujímalo, akú časť hry prespala. Na základe ofarbeného poľa zistite, koľko najmenej kôl muselo prebehnúť od začiatku hry do momentu, kedy Paulínka prišla.

Úloha

Dostanete mriežku rozmerov $r \times s$, ktorá je na každom políčku ofarbená jednou farbou. Táto mriežka bola na začiatku hry čistá. Potom prebehlo k kôl. V každom kole sa prefarbil práve jeden celý stĺpec alebo riadok tejto mriežky jednou farbou. Na základe výsledného ofarbenia zistite, koľko najmenej kôl mohlo prebehnúť, teda minimálne možné k .

Formát vstupu

V prvom riadku vstupu sú dve čísla: r, s ($1 \leq r, s \leq 200$), počet riadkov a stĺpcov mriežky. Každý z nasledujúcich r riadkov obsahuje s čísel oddelených medzerami f_i ; $0 \leq f_i \leq r \cdot s$, ktoré popisujú farby jednotlivých políčok mriežky.

Farby používajú celý rozsah čísel na vstupe. Inými slovami, ak sa vo vstupe vyskytuje farba číslo f , určite sa vo vstupe nachádzdzajú aj farby $0, 1, \dots, f - 1$.

Vo vstupoch je tiež garantované, že tabuľka vznikla podľa pravidiel hry.

Formát výstupu

Na výstup vypíšte jedno číslo k , minimálny počet ofarbení stĺpca/riadku potrebných na dosiahnutie ofarbenia tabuľky. ## Hodnotenie Vaše riešenia budú testované na štyroch sadách testovacích vstupov, pre ktoré platí:

Sada	1	2	3	4
Maximálne $r + s$	10	20	200	300

Príklady

vstup

```
2 5
0 0 1 1 1
0 0 1 1 1
```

výstup

```
4
```

V optimálnom prípade ofarbíme najprv oba riadky farbou 1 a potom prvé dva stĺpce farbou 0.

vstup

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

výstup

```
7
```

Ofarbujeme postupne: tretí riadok farbou 6, druhý stĺpec 5, druhý riadok 4, štvrtý stĺpec 3, posledný riadok 2, prvý stĺpec 1, prvý riadok 0.

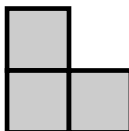
5. Prestavba bytu

12 b za popis, 8 b za program

Krtko rekonštruje byt. Posledné, čo mu zostáva je vydláždiť novú kúpeľňu. Zhodou náhod sa mu v schránke objavil leták obľúbeného veľkoskladu - Kúpeľňového Sveta Podláh - a v ňom zľava na dlaždice L-kového tvaru. Táto ponuka bola natoľko lákavá, že Krtko nakúpil oveľa viac dlaždíc, ako potreboval. Teraz sedí pred svojou dokonalou dokonalo-štvorcovou kúpeľňou a premýšľa, či on vôbec bude vedieť pomocou týchto kachličiek svoju kúpeľňu vydláždiť. Akoby nestačilo, v jeho inak príjemne dokonalo-štvorcovej kúpeľni sa nachádza otvor na odtok sprchy, ktorý nemôže kachličkami prekryť. Pomôžte Krtkovi vydláždiť jeho kúpeľňu alebo zistite, že to nie je možné.

Úloha

Štvorcovú kúpeľňu si môžeme predstaviť ako štvorcovú sieť $n \times n$. Pre dokonalo štvorcovú kúpeľňu navyše platí, že veľkosť jej strany n je mocnina dvojky. Odtok zaberá v myslenej sieti práve jedno celé políčko. Pod dlaždicou L-kového tvaru myslíme tri štvorce veľkosti 1×1 , spojené do tvaru L.



Vykachličkovanie je možné, ak pomocou L-kových kachličiek vieme prekryť všetky voľné miesta v našej kúpeľni (okrem odtoku) bez toho, aby sme nejaké kachličky akokoľvek delili alebo ľubovoľne prekryvali. Taktiež sa nesmie stať, že by nejaká kachlička vyčnievala z kúpeľne. Samozrejme, nakúpené kachličky je možné akokoľvek otáčať. Kachlička teda môže byť orientovaná 4 rôznymi spôsobmi.

Na vstupe od nás dostanete veľkosť kúpeľne. Ďalej od nás dostanete pozíciu odtoku. Ak je možné kúpeľňu takýchto rozmerov s odtokom na danom mieste vykachličkovať iba pomocou nakúpených L-kových kachličiek, vypíšte ľubovoľné takéto vykachličkovanie. Ak to možné nie je, podajte o tom správu.

Formát vstupu

Na prvom riadku vstupu sa nachádza číslo n ($1 \leq n \leq 256$) – dĺžka hrany štvorca reprezentujúceho kúpeľňu, pričom platí, že n je mocnina 2. V ďalšom riadku nasleduje dvojica čísel r, s ($1 \leq r, s \leq n$) – číslo riadku a stĺpca v našej mriežke, kde sa nachádza odtok.

Formát výstupu

Ak vykachličkovanie nie je možné, vypíšte na jeden riadok výstupu reťazec **Neda sa**. Inak si označme všetky použité kachličky číslami od 1 do x , kde x je ich počet (je zrejmé, že $x = (n^2 - 1) \div 3$). Na výstup vypíšte n riadkov. V každom takomto riadku sa má nachádzať n medzerou oddelených reťazcov. Ako j -ty reťazec v i -tom riadku vypíšte číslo kachličky, ktorá prekryva políčko v i -tom riadku a j -tom stĺpci mriežky alebo vypíšte znak X ak sa na danom políčku nachádza odtok. Korektných riešení môže byť viac (napríklad prečíslovaním kachličiek dostaneme z jedného korektného riešenia iné korektné riešenie). Vypíšte ktorékoľvek z nich.

Hodnotenie

Vaše riešenia budú testované na štyroch sadách testovacích vstupov, pre ktoré platí:

Sada	1	2	3	4
$1 \leq n \leq$	4	16	64	256

Príklady

vstup	výstup
<pre>2 1 1</pre>	<pre>X 1 1 1</pre>

Kúpeľna veľkosti 2×2 s odtokom v ľavom hornom rohu. Je len jedno možné riešenie.

vstup	výstup
<pre>4 2 3</pre>	<pre>2 2 1 1 2 3 X 1 4 3 3 5 4 4 5 5</pre>

Jedno z možných riešení. Akékoľvek iné riešenie s iným prečíslovaním či inou pozíciou kachličiek bude tiež akceptované.

6. Optimalizácia písania na klávesnici

12 b za popis, 8 b za program

Samko sa nedávno zamestnal ako programátor pre jeden veľký webový portál. Už počas prvého týždňa zistil, že to nebude jednoduchá práca. Má totiž na starosti opravovanie starého zdrojového kódu s množstvom chýb. Jeho bežný pracovný deň vyzerá tak, že zapne počítač, otvorí si mail a nájde desiatky škaredých správ od frustrovaných užívateľov. Tí sa väčšinou oprávnene sťažujú na množstvo bugov, ktoré sa na stránke nachádzajú. Samko samozrejme musí všetky takéto prípady preveriť, bugy nájsť a opraviť. Preto sa pomocou `ssh` na diaľku pripojí k príkazovému riadku (terminálu) servera a prehľadávaním logov a zdrojového kódu zisťuje, kde nastala chyba.

Keďže práce je veľa, Samko sa snaží využívať svoj čas efektívne. Napríklad sa naučil bezchybne písať všetkými desiatimi prstami. Teraz však objavil novú funkciu jeho terminálu – slovník často používaných slov (príkazov). Keď do terminálu napíše začiatok nejakého slova zo slovníku a stlačí `TAB`, tak sa mu tam toto slovo celé dopíše. Ak sa viac slov v slovníku začína rozpísaným slovom, tak sa dopíše lexikograficky najmenšie z nich. Ak žiadne slovo v slovníku nezačína rozpísaným slovom, tak v tom prípade kláves `TAB` nespraví nič. Samko po dlhej úvahe dospel k tomu, že táto funkcia mu vie ušetriť veľa zbytočných stlačení klávesnice. Napríklad ak má v slovníku lexikograficky najmenšie slovo `auto` a chce napísať slovo `aut`, tak to môže spraviť tak, že stlačí `TAB` a `Backspace` (kláves na zmazanie posledného znaku). Namiesto troch stlačení tak spraví iba dve.

Samka by zaujímalo, koľko času mu táto funkcia vie ušetriť. Preto by si chcel spraviť štatistiku o tom, koľko stlačení potrebuje na napísanie určitých slov. Keďže má však ešte veľa roboty s hľadaním bugov, potreboval by, aby mu s tým niekto pomohol. Zvládnete to?

Úloha

Na vstupe je terminálový slovník často používaných slov a zoznam slov, ktoré Samka zaujímajú. Pre každé slovo zo Samkovho zoznamu vypíšte, koľko stlačení klávesnice je potrebných na jeho napísanie. Samkov terminál nepodporuje používanie šípok a iných kláves na presúvanie kurzora. Jediné klávesy, ktoré Samko používa sú písmená, `TAB` a `Backspace` (na zmazanie posledného znaku).

Formát vstupu

Na prvom riadku vstupu sa nachádzajú čísla n a q , – prvé určuje počet slov v slovníku ($1 \leq n \leq 1\,000\,000$) a druhé počet slov v Samkovom zozname ($1 \leq q \leq 1\,000\,000$). V každom z nasledujúcich n riadkoch je jedno slovo zo slovníka. Na ďalších q riadkoch sa nachádzajú postupne slová v Samkovom zozname. Všetky slová na vstupe sa skladajú z malých písmen anglickej abecedy a majú menej ako 60 000 písmen. Celkový počet písmen na vstupe nepresiahne 2 150 000.

Formát výstupu

Pre každé z q slov v Samkovom zozname vypíšte jeden riadok – najmenší možný počet stlačení kláves na napísanie daného slova.

Príklad

vstup

```
3 7
autobus
auto
autostrada
auto
aut
a
autobusar
autobus
autob
utobus
```

výstup

```
1
2
1
5
3
2
6
```

Všimnite si, že slovo **auto** je v slovníku lexikograficky najmenšie. Preto na jeho napísanie stačí jedno stlačenie **TAB**.

Slovo **autobusar** vieme napísať nasledujúcov postupnosťou kláves: **TAB, b, TAB, a, r**.

Slovo **utobus** musíme napísať celé znak po znaku.

7. Daj het

4 b za popis, 16 b za program

Máme štvorcovú tabuľku. V tabuľke máme prirodzené čísla. Pekne spôsobne, v každom políčku jedno.

Každý riadok aj každý stĺpec má pri sebe napísané jedno číslo. Toto číslo hovorí, aký súčet chceme v danom riadku/stĺpci dosiahnuť.

Úloha

Daj het niektoré čísla z tabuľky tak, aby všetky súčty sedeli.

Formát vstupu

Toto je open data úloha, dostanete teda od nás úplne všetky vstupy. Môžete sa na ne ľubovoľne pozeráť a robiť s nimi čo len chcete. Vstupov je 8 a volajú sa 1.in až 8.in. Každý vstup začína číslom t predstavujúcim počet testov, ktoré obsahuje. Každý test má nasledujúci formát:

- V prvom riadku je číslo n : rozmer tabuľky. (V každom vstupe majú všetky testy rovnaké n .)
- V druhom riadku je n čísel: predpísané súčty pre riadky zhora dole.
- V treťom riadku je n čísel: predpísané súčty pre stĺpce zľava doprava.
- V každom z nasledujúcich n riadkov je n kladných celých čísel nepresahujúcich 20, v poradí, v akom sú rozmiestnené v tabuľke.

Môžete predpokladať, že pre každú zo zadaných tabuliek existuje aspoň jedno riešenie.

Vstupy pre [windows tu](#)², pre [linux tu](#)³.

Formát výstupu

Odovzdajte zip archív a v ňom príslušné výstupy. Tie musia byť pomenované 1.out až 8.out a umiestnené v koreňovom adresári zipu. Za každý výstup, ktorý sa v zipe naozaj bude nachádzať a bude úplne správny, dostanete dva body.

Formát výstupu: pre každý test vypíšete n riadkov a v každom n čísel: čísla vo výslednej tabuľke, pričom namiesto čísel, ktoré ste “dali het” (čítaj: z tabuľky odstránili) vypíšete nuly.

Môžete vypisovať aj nadbytočné medzery a voľné riadky medzi výstupmi pre jednotlivé testy, náš testovač ich spokojne odignoruje. Ak existuje pre nejakú tabuľku viacero riešení, môžete vypísať ľubovoľné jedno z nich.

Posledné nanajviš 4 body za úlohu udelíme za popis toho, ako ste úlohu riešili, a obzvlášť programu/ov, ktoré ste použili. Rozumne podrobný popis ľubovoľného postupu, ktorý nejaké vstupy naozaj vyriešil, môže dostať aspoň 3 body.

²http://media.ksp.sk/ulohy/36rocnik/2kolo/7vstupy_windows.zip

³http://media.ksp.sk/ulohy/36rocnik/2kolo/7vstupy_linux.zip

Príklady

vstup

```
1
5
15 12 14 14 24
10 20 18 14 17
7 3 2 5 1
3 5 4 3 3
5 3 3 9 5
5 6 4 3 4
4 9 8 8 7
```

výstup

```
7 0 2 5 1
3 5 4 0 0
0 0 0 9 5
0 6 4 0 4
0 9 8 0 7
```

8. Svinstvo na koberci

12 b za popis, 8 b za program

Krtko si na oslavu svojho novo zrekonštruovaného bytu pozval všetkých vedúcich. Kubíkovi sa byt tak páčil, že sa rozhodol kúpiť nový nábytok do Klubovne Spokojných Programátorov. Zohnal nové staré stoly, kolieskové stoličky bez koliesok aj pekné preplnené skrine. Nákup zavŕšil novým viacdielnym kobercom. Ten sa skladá z viacerých kúskov uložených tak, že na podlahe tvoria mapu.

Netrvalo dlho a na čistučkome koberci sa začali každý deň objavovať zablatené stopy, zadupaná čokoláda či škrny od kofoly. Tie treba rýchlo odstrániť, inak sa do koberca zažerú nastálo.

Kubík dostal briliantný nápad. Na strop pripevnil kameru namierenú na koberec, takže vie presnú polohu každej škvrny v momente, keď sa objaví. Od vás chce program, ktorý mu povie veľkosť zašpineného kusu koberca, aby mohol objednať správne množstvo čistiacich prostriedkov.

Úloha

Na vstupe dostanete zadanú mapu koberca v podobe **súvislého planárneho grafu** (vrcholy so súradnicami a hrany medzi nimi) a niekoľko bodov. Vašou úlohou je pre každý bod nájsť veľkosť uzavretej oblasti, v ktorej leží (tj. jej obsah). Ak sa bod nenachádza v žiadnej uzavretej oblasti, vypíšte -1 . **Body ležiace na niektorej hrane grafu (vrátane jej okrajov) považujeme, že nie sú vo vnútri žiadnej uzavretej oblasti.**

Pozor! Túto úlohu je treba riešiť online, čiže testovač vášmu programu posiela otázky postupne. Každú nasledujúcu otázku dostanete až po tom, čo zodpoviete aktuálnu. Nezabudnite flushovať, aby sa vaša odpoveď skutočne testovaču odoslala a netrávila prázdny v nejakom buffri. (C++ `cout.flush()`, Python `sys.stdout.flush()`, pre iné jazyky si vyhľadajte.)

Formát vstupu

Na prvom riadku vstupu je číslo n —počet vrcholov, m —počet hrán a q —počet otázok.

Potom prázdny riadok, a nasleduje n riadkov ($n \geq 1$) obsahujúcich 2 čísla—súradnice daného vrchola. Môžete predpokladať, že žiadne dve vrcholy nie sú ten istý bod v rovine. Všetky súradnice sú celé čísla, ktoré v absolútnej hodnote neprevýšia 10^9 .

Potom prázdny riadok, a na ďalších m riadkoch ($m \geq 0$) sú vždy 2 čísla—čísla vrcholov, medzi ktorými vedú hrany. Vrcholy číslujeme $1, 2, \dots, n$. Môžete predpokladať, že ak niektoré tri vrcholy a, b, c ležia na jednej priamke v tomto poradí, tak na vstupe nebude hrana ac . Takisto sa žiadna hrana na vstupe nevyskytne viackrát, a žiadne dve hrany sa nekrižujú. Koncové vrcholy každej hrany sú rôzne. Zadaný graf je súvislý.

Potom prázdny riadok, a na každom z posledných q riadkov ($q \geq 0$) sú 2 čísla—súradnice bodu z otázky.

Súradnice všetkých bodov sú nezáporné a neprevyšujú 10^9 .

Formát výstupu

Pre každý bod vypíšte riadok obsahujúci jedno číslo—veľkosť oblasti, v ktorej sa nachádza. Ak bod neleží v žiadnej uzavretej oblasti, vypíšte -1 . **Pokiaľ je výsledok celočíselný, nevypisujte žiadne desatinné miesta, v opačnom prípade vypíšte jedno desatinné miesto.** (Rozmyslite si, že obsah každej z oblastí je nutne celočíselným násobkom $\frac{1}{2}$.)

Hodnotenie

Obmedzenia v jednotlivých sadách sú nasledovné:

Sada	1	2	3	4	5	6	7
$n \leq$	300		2 500			50 000	
$m \leq$	300		4 000			70 000	
$q \leq$	10 000					20 000	

Príklady

vstup

```

5 7 5

10 6
7 10
9 1
0 6
2 0

2 4
5 3
1 3
3 4
5 4
4 1
2 1

8 6
0 3
2 5
7 0
8 1

```

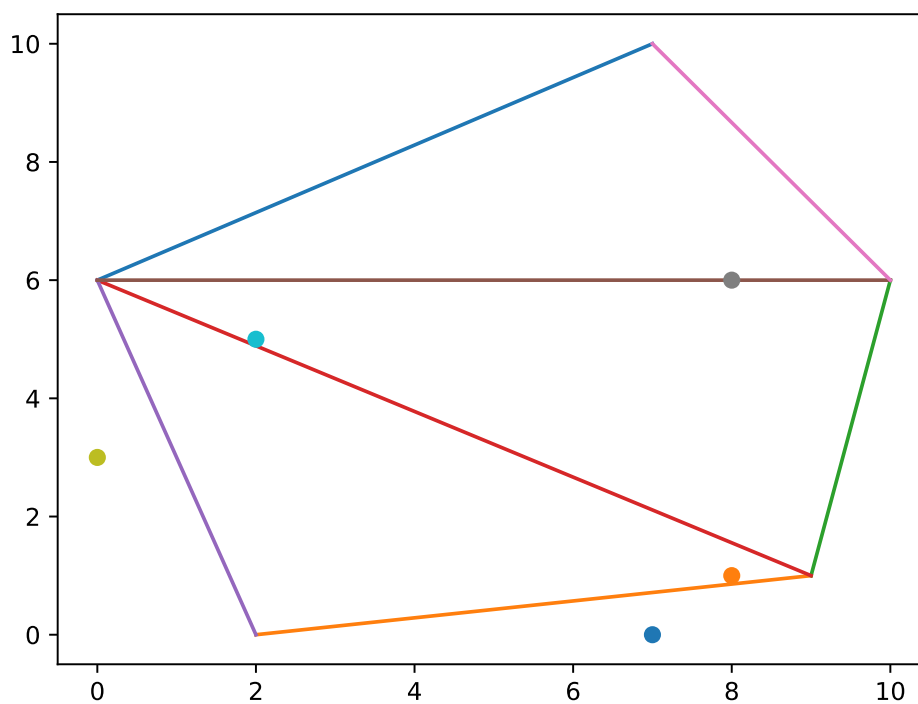
výstup

```

-1
-1
25
-1
22

```

Obrázok prvého vstupu:



vstup

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

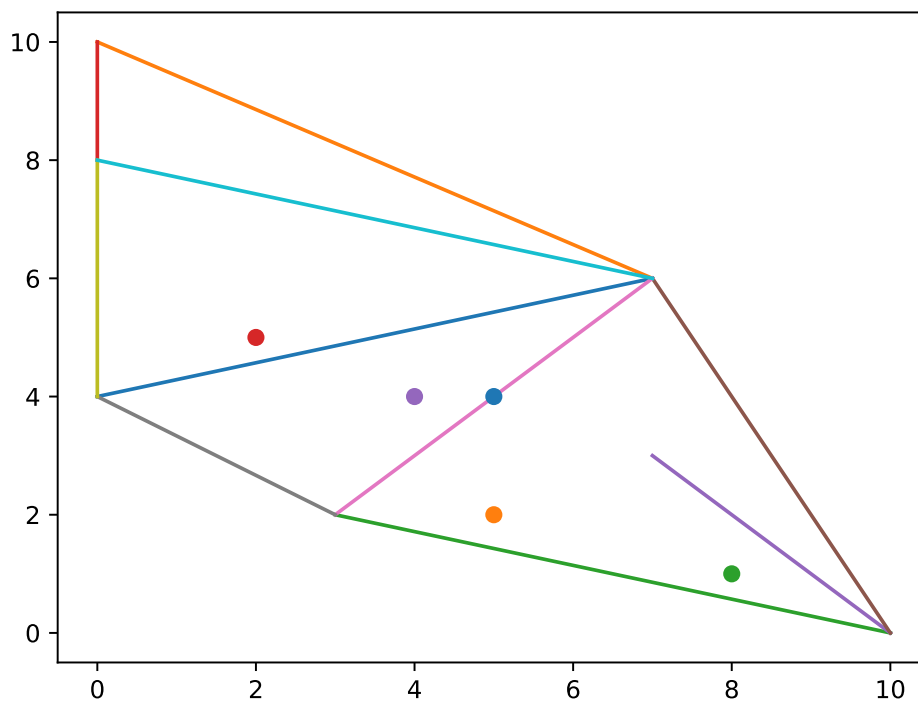
6 2
6 7
5 4
3 7
5 1
6 5
6 4
2 4
3 2
6 3

5 4
5 2
8 1
2 5
4 4
```

výstup

```
-1
18
18
14
10
```

Obrázok druhého vstupu:



vstup

```
9 12 5
3 1
4 5
7 0
10 5
8 0
0 6
1 10
5 5
6 5

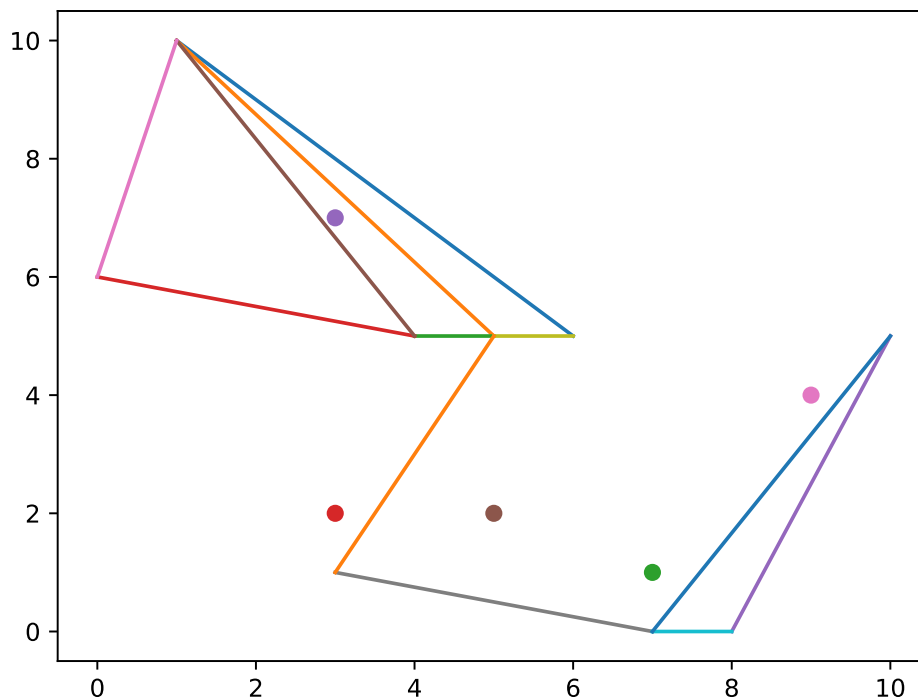
9 7
8 7
8 2
2 6
4 5
7 2
6 7
3 1
8 9
3 5
3 4
1 8

7 1
3 2
3 7
5 2
9 4
```

výstup

```
-1
-1
2.5
-1
-1
```

Obrázok tretieho vstupu:



vstup

```
5 5 5
0 0
0 4
4 4
4 0
1 1

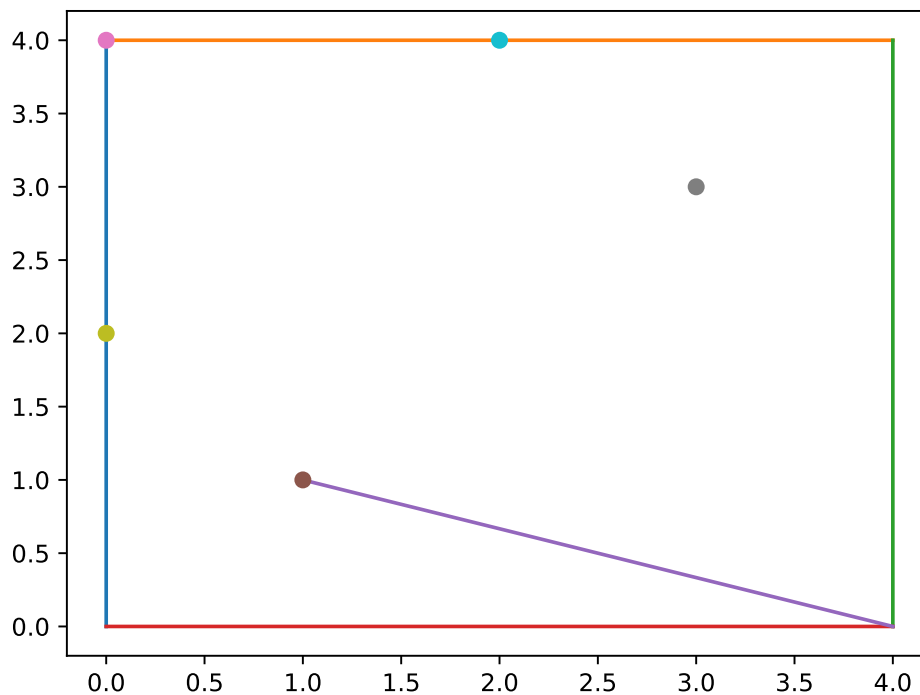
1 2
2 3
3 4
4 1
4 5

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

výstup

```
-1
-1
16
-1
-1
```

Obrázok štvrtého vstupu:



Pozastavenie kategórie T

Kategória T tento semester prebiehať **nebude**. Ak vám nestačia úlohy KSP, vyskúšajte aj iné súťaže a programátorské stránky. Odkazy na niekoľko veľmi dobrých stránok a súťaží podobného štýlu ako KSP nájdete na <https://www.ksp.sk/trening/>.