



Úlohy 1. kola letnej časti

Termín odoslania riešení tohto kola je pondelok **27. 5. 2019**. Doprogramovanie končí v pondelok 10. 6. 2019.

1. Párty

11 b za popis, 8 b za program

Kika peče tortu. Narodeninová párty jej sestry je už čoskoro a ona ešte len dopiekla cesto, resp. veľa ciest, keďže jej sestra je ešte malá a chce vysokú tortu. Dokonca nie len tak nejakú, ale viacfarebnú. Kika teda napiekla niekoľko čokoládových a vanilkových ciest. Teraz stojí pred neľahkou úlohou a to poukladať ich na seba tak, aby to bolo dostatočne pestrofarebné a rozmanité, ale nie veľmi prehnané. Všimla si, že najvýraznejšie sú miesta, kde prechádza farba jedného cesta na druhú farbu. Zadefinovala si teda prechod ako miesto, kde sa stretávajú cestá dvoch rôznych farieb. Chcela by vidieť, ako vyzerá torta pre rôzne počty prechodov. Pomôžte jej.

Úloha

Kika by chcela program, ktorý pre zadaný počet čokoládových ciest, vanilkových ciest a počet prechodov vytvorí jeden konkrétny vzhľad torty.

Formát vstupu

Na jedinom riadku vstupu sú tri medzerou oddelené čísla $1 \leq c \leq 500\,000$, $1 \leq v \leq 500\,000$ a $1 \leq p \leq 1\,000\,000$ pričom c označuje počet čokoládových ciest, v počet vanilkových ciest a p počet prechodov, ktoré má mať výsledná torta.

Formát výstupu

Na jediný riadok výstupu vypíšte jeden reťazec dĺžky $c + v$ znakov, pričom práve c z nich bude c a v z nich v . Okrem toho má platiť, že počet prechodov, teda dvojíc cv a vc , má byť dokopy presne p . Za týmto reťazcom má už nasledovať iba znak konca riadka. Je garantované, že riešenie **vždy existuje**. Pokiaľ existuje viac korektných riešení, vypíšte ľubovoľné z nich.

Príklad

vstup

5 1 2

výstup

cccvc

Výsledná torta má mať 5 čokoládových ciest a 1 vanilkové. Počet prechodov v tejto torte je 2 a to z 3. vrstvy na 4. vrstvu a tiež zo 4. na 5. Existujú, samozrejme, aj iné korektné výstupy ako cvcccc či ccccvc. Príkladom zlého riešenia je vcccc, pretože počet prechodov je iba 1 a to z 1. na 2. vrstvu.

2. Riadny strach

12 b za popis, 8 b za program

Krtko a Jerry sa idú zapísať na ústnu skúšku z analýzy. Keďže analýza je najväčší strašiak matfyzákov, chceli by sa na skúške navzájom podporovať svojou prítomnosťou. Vedia, že profesor vždy volá študentov na ústnu skúšku po skupinkách. Najväčšia šanca, že budú v jednej skupinke je, ak sa zapíšu hneď po sebe. Na skúšku sa však musia poctivo pripravovať, preto nemajú čas zisťovať, na ktoré miesto sa majú zapísať aby išli s čo najväčšou pravdepodobnosťou spolu. Pomôžete im?

Úloha

Na skúšku je voľných n miest. Zistite, aká je pravdepodobnosť, že Krtko a Jerry pôjdu spolu na skúšku ak sa Jerry zapíše na r -tú pozíciu a Krtko hneď pred ňu. Profesor si vždy pred skúškou náhodne zvolí x – veľkosť skupiniek po ktorých bude študentov volať dnu, pričom platí, že $1 \leq x \leq n$. Každá veľkosť skupinky od 1 po n je rovnako pravdepodobná.

Formát vstupu

Na vstupe sa nachádza jeden riadok a v ňom dve čísla: n - počet ľudí zapísaných na danú skúšku a r - poradie Jerry, pričom platí ($2 \leq r \leq n \leq 10^{12}$). Pozor, v jazykoch podobných C++ vám nemusí na uloženie n a r stačiť klasický typ `int`. Riešením (v C++) je napr. použiť typ `long long`.

V testovacích sádach platia pre n a r nasledovné obmedzenia zhora:

Sada	1 a 2	3	4	5	6 až 8
$n, r \leq$	10 000	100 000	150 000 000	250 000 000	10^{12}

Formát výstupu

Vypíšte dve medzerou oddelené čísla p a q kde p je čitateľ a q menovateľ zlomku, ktorý vyjadruje pravdepodobnosť s akou pôjdu Krtko s Jerry odpovedať v jednej skupinke. Zlomok navyše vypíšte v **základnom tvare**.

Príklad

vstup	výstup
20 15	4 5

Na skúšku je zapísaných 20 ľudí pričom Krtko je v poradí 14. a Jerry 15. Krtko a Jerry budú spolu v skupinke ak bude profesor volať študentov po skupinkách veľkosti 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19 alebo 20. Celkovo profesor môže vymyslieť 20 rôznych veľkostí skupiniek (1-20), preto je pravdepodobnosť, že pôjdu spolu $16/20$, teda v základnom tvare $4/5$.

3. Otravné notifikácie

12 b za popis, 8 b za program

Baklažán má konečne nový mobil. A nie hocijaký. Je to novučičký KSP Nokia Lumia S6, limitovaná edícia, s umelou inteligenciou na každom pixeli. Konečne totiž pochopil, že tie smartfóny sú v skutočnosti celkom užitočné, keď mu na starom mobile došla pamäť pri vkladaní nového kontaktu.

Jeho nový mobil sa mu ale bohužiaľ snaží dokázať opak: vyhadzuje jednu notifikáciu za druhou, s náhodnou informačnou hodnotou. Nemôže ich všetky ignorovať, lebo čo ak tá nasledujúca bude dôležitá? Už mu dochádza trpezlivosť... Koľko notifikácií ešte bude musieť dnes prečítať?

Úloha

Na začiatku nie sú na mobile žiadne neprečítané notifikácie. Následne prichádzajú tri typy udalostí:

1. Aplikácia x vygenerovala jednu novú (neprečítanú) notifikáciu.
2. Baklažán prečítal všetky (neprečítané) notifikácie od aplikácie x .
3. Baklažán prečítal prvých t neprečítaných notifikácií. Môžete predpokladať, že Baklažán má aspoň toľko neprečítaných notifikácií, ale môže ich mať aj viac.

Po každej udalosti vypíšte, koľko notifikácií je ešte neprečítaných. (Do toho sa nerátajú notifikácie, ktoré ešte neboli vygenerované.)

Formát vstupu

Na prvom riadku vstupu sú dve čísla oddelené medzerou: počet aplikácií n a počet udalostí q . Platí $1 \leq n, q \leq 500\,000$.

Každý z nasledujúcich q riadkov popisuje jednu udalosť. Popis udalosti má nasledovný formát: $1 \langle x \rangle$, $2 \langle x \rangle$ a $3 \langle t \rangle$ postupne pre udalosti prvého, druhého a tretieho typu. Platí $1 \leq x \leq n$ (aplikácie číslujeme od 1) a $1 \leq t$.

Formát výstupu

Vypíšte q riadkov, na i -tom z nich nech je počet zostávajúcich notifikácií po i -tej udalosti.

Hodnotenie

Sú 4 testovacie sady, každá za 2 body. Platia v nich nasledovné obmedzenia:

Sada	1	2	3	4
$n, q \leq$	500	5 000	50 000	500 000

Príklad

vstup

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

výstup

```
1
2
3
2
```

vstup

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

výstup

```
1
2
1
1
```

Pri tretej udalosti Baklažán prečíta notifikáciu aplikácie 1, nakoľko tá je prvá v poradí. Pri štvrtej udalosti nič neprečíta, lebo aplikácia 1 už nemá žiadne ďalšie notifikácie.

vstup

```
4 7
1 2
1 4
1 2
2 3
1 3
3 1
3 1
```

výstup

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

Pri štvrtej udalosti Baklažán nič neprečíta, lebo sa pozerá na aplikáciu 3, ktorá vtedy ešte nič nevygenerovala.

4. Treba sa najesť

12 b za popis, 8 b za program

V krajine, kde sa piesok lial a voda sypala, je jazierko s radom kameňov. Kamene sú v poradí označené písmenami reťazca S – prvý kameň má písmeno $S[0]$, druhý $S[1]$ atď. Na každom kameni je kúsok pizze. Žabiak Žaba je hladný a preto chce preskákať cez čo najviac kameňov. Nie je to ale také jednoduché, pretože na rozdiel od žiab v normálnom svete nemôže skočiť do vody, v sypajúcej vode totiž plávať nevie. Vie sa však po brehu jazera vrátiť zase na začiatok a skákať po kameňoch odznova. Kamene sú navyše príliš malé na to, aby sa na nich otáčal, čiže nemôže skákať na predchádzajúce kamene, je ale dostatočne silný na to, aby doskočil na kameň ľubovoľne ďaleko.

Problém je ešte zapeklitejší, pretože v krajine práve vládne krutovládca s menom T a ten vydal rozkaz, že ak niekto nebude skákať po kameňoch, ktoré tvoria reťazec T , tak ich roztrhne ako žabu. Keby Žaba nebol prototypom dobrého občana, všetko by bolo jednoduchšie. Preskákala by postupne po každom kameni a dúfala, že si ho nikto nevšimne. To však nie je ochotný spraviť, a preto teraz rozmýšľa, či vie skočiť na každý kameň v jazierku a pritom neporušiť vládcov rozkaz.

Úloha

Máme dva reťazce, S a T . Z reťazca S vieme vytvoriť podreťazec tak, že z neho vyškrtujeme niektoré písmená a zvyšné zachováme v pôvodnom poradí. Zoberme si všetky podreťazce S , ktoré sú zhodné s T . Zistite, či každý znak reťazca S patrí aspoň do jedného takéhoto podreťazca.

Formát vstupu

Na prvom riadku vstupu je číslo udávajúce počet testovacích zád. Sád je najviac sto. Nasledujú dva riadky pre každú sadu, na ktorých sú reťazce S a T . Reťazce pozostávajú z malých písmen anglickej abecedy a nie sú prázdne. Dĺžka každého reťazca nepresiahne 10^5 a dĺžka všetkých reťazcov v jednom vstupe nepresiahne 10^6 .

Formát výstupu

Vypíšte ano, ak sa každý znak reťazca S nachádza aspoň v jednom z podreťazcov tvoriacich slovo T . Inak vypíšte nie.

Hodnotenie

Sú 4 testovacie sady, každá za 2 body. Platia v nich nasledovné obmedzenia:

Sada	1	2	3	4
dĺžka každého reťazca \leq	100	1 000	10 000	100 000

Príklad

vstup

```
3
haha
ha
zabajezaba
zaba
kamen
ak
```

výstup

```
ano
nie
nie
```

5. Ooo, Vlaky

12 b za popis, 8 b za program

Píše sa rok 2143. Presne pred 100 rokmi sa človek stal multiplanetárnym druhom, keď prví vesmírni osadníci pristáli na Marse a následne vybudovali kolóniu. Na začiatku ich tam bolo presne 157 a každý tak mal dostatok priestoru, keďže pôvodná kapacita prvej kolónie sa šplhala až k číslu 1223.

Časom však populácia červenej planéty rástla¹. 53% terajších obyvateľov pricestovalo zo Zeme alebo z Mesiaca a 47% sa na Marse už narodilo. Preto už takýchto kolónií existuje niekoľko (rádovo v stovkách). Čoskoro, teda, dôjde k vyrovnaniu pomeru úplných² a polovičných³ “Marťanov”, a veľkosť populácie prekročí 200 000.

Znie to veľmi sľubne a prosperujúco, no obyvateľov Marsu stále ešte obmedzuje pár vecí, medzi ktoré patrí napríklad aj doprava. Na prepravu materiálu, potravín alebo osôb sa v tomto nehostinnom teréne používajú elektrické rovery. Tie majú svoje nevýhody: sú pomalé, často sa kvôli prašnému povrchu kazia, majú relatívne malú nosnosť a, samozrejme, z bezpečnostných dôvodov nepremávajú počas Marťanskej búrky. Preto sa Rada kolónií Marsu, Aliancia štátov pre multiplanetárny život a Medzinárodná rada vesmírnych agentúr rozhodli riešiť dopravu na tejto planéte.

Za najvhodnejší spôsob prepravy osôb a tovaru tu považujú vlaky, premávajúce hermeticky uzavretými podzemnými tunelmi. Tie budú môcť premávať aj počas búrok⁴, uvezú veľa nákladu, nebudú vystavené prašnému prostrediu a budú relatívne rýchle⁵. Okrem toho, odhlasovalo sa, že z bezpečnostných dôvodov⁶ budú všetky tunely vybudované v rovnakej hĺbke, a to znamená, že žiadne dva tunely sa nesmú križovať. V každom tuneli bude premávať práve jeden vlak. Vlaky sú totiž najdrahšou položkou v tomto projekte a preto ich počet, a teda aj počet tunelov, bude čo najnižší. Zároveň ich treba tolko, aby sa bolo možné dostať z ľubovoľnej kolónie do hociktorej inej len použitím vlakov. Ak si teda vlakovú sieť predstavíme ako graf, tak bude vyzeráť ako strom, z čoho vyplýva, že počet tunelov bude počet kolónií mínus 1.

Plán projektu sa skladal z troch tabuliek:

- V jednej boli spárované identifikačné čísla pozícií s ich súradnicami na mape⁷.

¹Posledné informácie hovoria o počte 197 359

²Narodili sa už na Marse

³Pricestovali na Mars

⁴Keďže budú chránené v podzemí

⁵1373 km/h

⁶Ani predseda Medzinárodnej rady vesmírnych agentúr nevedel zdôvodniť toto rozhodnutie

⁷Aj keď je jasné, že Mars nie je plochý, pre účely tejto úlohy postačí, že povrch Marsu budeme považovať za rovinu a použijeme obyčajnú karteziánsku súradnicovú sústavu.

Pozícia	ID pozície
[0, 0]	1
[1, 1]	2
[2, 3]	3

- V druhej boli pozície priradené číslam a názvom kolónií.

ID pozície	ID kolónie	Kolónia
1	1	Jablkovo
2	2	Hruškovo
3	0	Malinovo

- V tretej sa nachádzal zoznam dvojíc čísel kolónií, ktoré je potrebné prepojiť tunelom.

ID kolónie a	ID kolónie b
1	2
0	2

Po pár mesiacoch bol plán projektu hotový a išlo sa stavať, no nastal problém. Druhá tabuľka sa záhadne stratila, a tak sú v projekte len súradnice a prepojenia čísel kolónií. Nevieme ale, ktorá kolónia je na ktorých súradniciach.

Stavebníci teraz nevedia, kde treba stavať. Skúste ale z projektu zachrániť, čo sa dá: navrhните také riešenie, ktoré teoreticky môže byť pôvodne zamýšľaný plán.

Úloha

Na Marse sa nachádza n kolónií. Poznáte ich n celočíselných súradníc x, y , avšak, nevíete konkrétne, ktorá pozícia prislúcha ktorej kolónii. Pozície sú navzájom rôzne, a žiadne tri pozície neležia na jednej priamke.

Ďalej máte plán projektu, v ktorom sú zapísané dvojice čísel a, b označujúce kolónie, medzi ktorými má viesť obojsmerný tunel. Táto vlaková sieť tvorí graf-strom: je teda súvislá a vedie v nej práve $n - 1$ hrán.

Navrhните, medzi ktorými pozíciami majú viesť tunely tak, aby sa žiadne dva tunely nekrižovali, a aby jednotlivé pozície v nejakom poradí zodpovedali jednotlivým kolóniám.

Vstup

V prvom riadku vstupu sa nachádza číslo $1 \leq n \leq 1\,000$ udávajúce počet kolónií. Nasleduje $n - 1$ riadkov. V každom z nich sa nachádzajú dve čísla a, b hovoriace, že medzi kolóniami a a b má byť postavený tunel. Kolónie číslujeme $0, 1, \dots, n - 1$.

Ďalej nasleduje n riadkov. V i -tom z nich sa nachádzajú dve čísla x_i, y_i , súradnice i -tej pozície nejakej kolónie. Tieto súradnice v absolútnej hodnote nepresiahnu 10^9 .

Výstup

Na výstup vypíšte $n - 1$ riadkov. V každom z nich nech je dvojica čísel a, b , ktorá znamená, že medzi a -tou pozíciou a b -tou pozíciou má viesť tunel. Pozície číslujeme od 0 po $n - 1$.

Na poradí tunelov ani na poradí čísel a a b vo výstupe nezáleží. Ak je správnych odpovedí viac, vypíšte ľubovoľnú z nich.

Je zaručené, že každý vstup má platné riešenie.

Príklady

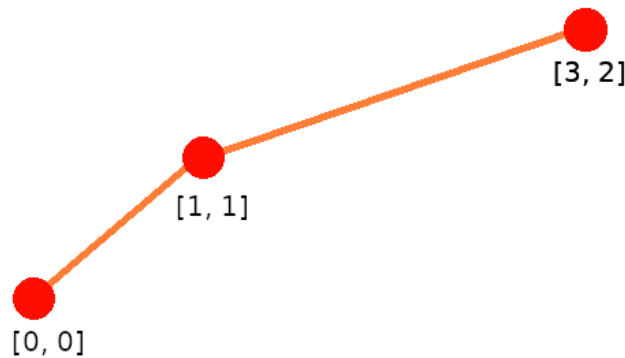
vstup

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

výstup

```
0 1
1 2
```

Máme tri kolónie. Očíslované sú 0, 1 a 2. Tunel má viesť medzi kolóniami 1 a 2 a medzi kolóniami 0 a 2. Tri pozície sú $[0, 0]$, $[1, 1]$ a $[2, 3]$. Príkladový výstup hovorí, že spojíme pozíciu číslo 0 s pozíciou 1 a tiež 1 s 2. To znamená, že jeden tunel bude medzi $[0, 0]$ a $[1, 1]$ a druhý medzi $[1, 1]$ a $[2, 3]$. Keby sme si to znázornili na štvorcovú sieť súradníc, videli by sme, že sa žiadne dva tunely nepretínajú. Okrem toho je zachovaný aj spôsob prepojenia kolónií: Buď by bola kolónia číslo 1 v $[0, 0]$ a kolónia 0 v $[2, 3]$, alebo naopak.



Iné platné riešenie by mohlo spojiť pozície 1 s 2 a 2 s 0

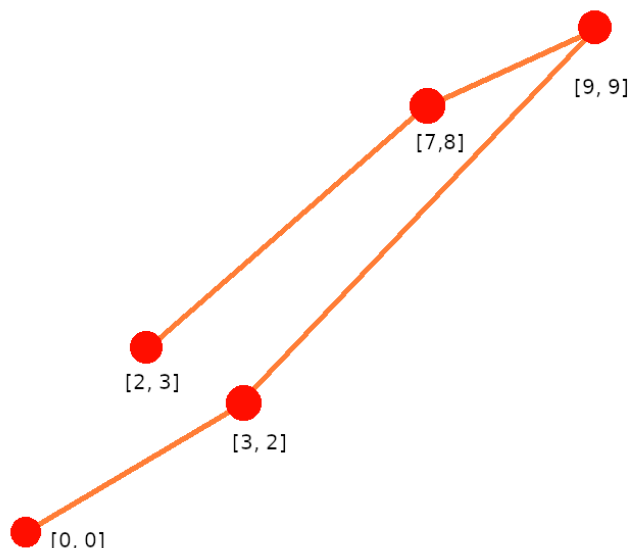
vstup

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

výstup

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

Podľa tohto výstupu by boli tunely postavené takto:



6. Totálna pohroma

12 b za popis, 8 b za program

Syseľ sa s hlbokým výdychom zvalil na svoju kancelársku stoličku. Hotovo! Po mesiacoch úpornej driny sa mu konečne podarilo dokončiť samofunkčný vysávač poháňaný jadrovým reaktorom. Jediný vysávač takéhoto druhu na celom svete. Proste unikát.

Bol položený na stole pred ním, chrómované telo odrážalo svetlo, trubica vedúca zo stredu sa pomaly otáčala a snažila sa nájsť špinu na vysatie, to všetko doplnené jemným bzukotom reaktora. A všade naokolo papiere, nákresy, výpočty, kusy kódu. Naprogramovať umelú inteligenciu bola pre Sysľa hračka, konštrukcia vysávača tiež nebola problematická. Reaktor mu však dal zabrať. Ani by ste netušili, aký je problém zohnať dostatočné množstvo kvalitného uránu. Ale ani keď ho len tak hodíte do vnútra, tak to nezačne fungovať. Lietajúce neutróny treba celý čas precízne kontrolovať, jednak aby jadrová reakcia nevyhasla, jednak aby to celé nevybuchlo.

Tajomstvom je použitie špeciálnej kontrolnej matice rozmerov $n \times n$, cez ktorú neutróny filtrujete. Každé políčko je buď priepustné alebo nepriepustné, je však dôležité, aby platilo, že v každom štvorci okolo hlavnej diagonály je práve párny počet nepriepustných políčok. Presnejšie, ak si ľavý horný roh matice označíte súradnicami $(1, 1)$ a pravý dolný súradnicami (n, n) , tak musí platiť, že každý štvorec, ktorého protíľahlé rohy sú na súradniciach (i, i) a (j, j) (pre $i < j$) má párne veľa nepriepustných políčok. Syseľ pohľadom zavadil o nákres tejto kontrolnej matice, ktorý ležal pred ním na stole. Asi by ho mal niekam odložiť, predsa je to len najdôležitejšia časť jeho výtvoru a on si ju ešte nestihol zálohovať.

V tom však do miestnosti vstúpila ona a Syseľ zabudol na všetko ostatné. Nikol. Najlepšia programátorka v celom VacuumLabs. Keby ste ju videli ako píše na svojej ergonomickkej klávesnici. Prsty sa jej len tak mihajú po klávesách. A keď potom spraví pull request, pozerať sa naňho chodia ešte aj z Exponei. K tomu všetkému je ešte aj nesmierne krásna a nezadaná. Len nedávno sa síce rozišla so svojim priateľom⁸, ale je najvyšší čas, aby Syseľ spravil prvý krok. A jadrový vysávač môže byť presne tá vec, ktorá ju ohúri.

Keď Nikol vošla do miestnosti, Syseľ sa v okamžiku postavil zo stoličky, aby ju privítal. Pritom však na stôl vysypal omrvinky z keksikov, ktoré jedol počas práce. Jeho inteligentný vysávač to hneď zaregistroval a začal otáčať trubicu aby ich vysal. Pritom však omylom prevrhol pohár vody a tá sa rozliala po nákrese kontrolnej matice. Katastrofa! Ako teraz Syseľ ohúri Nikol, keď je jeho výtvor zničený? Na mokrom papieri je vidno už iba niekoľko 0 a 1, ktoré predstavujú priepustné a nepriepustné políčka. Pomôžte Sysľovi zistiť, koľkými spôsobmi vie doplniť zvyšok matice, nech sa môže pustiť do opravovania.

Úloha

Na vstupe dostanete veľkosť matice n a políčka, ktoré ostali zachované spolu s ich pôvodnými hodnotami – 0 alebo 1. Zistite, koľkými spôsobmi je možné doplniť zvyšné políčka tejto matice číslami 0 a 1 tak, aby platilo, že v každom štvorci okolo hlavnej diagonály so stranou väčšou ako 1 je párne veľa jednotiek.

Hlavná diagonála matice je tvorená políčkami, pre ktoré platí, že ich x -ová a y -ová súradnica je rovnaká. Ťahá sa preto od ľavého horného rohu (súradnice $(1, 1)$) k pravému dolnému rohu (súradnice (n, n)). Štvorec je okolo hlavnej diagonály vtedy, ak dva jeho protíľahlé rohy ležia na hlavnej diagonále.

⁸Bol to hlupák, ak chcete počuť Sysľov názor.

Keďže počet spôsobov, ako doplniť maticu, môže byť naozaj veľký, vypíšte iba ich zvyšok po delení číslom 1 000 000 009.

Formát vstupu

Na prvom riadku vstupu sa nachádzajú dve celé čísla n a m ($2 \leq n \leq 10^5$, $0 \leq m \leq \min(50\,000, n^2)$) – veľkosť matice a počet zachovaných políčok.

Nasleduje m riadkov, každý popisuje jedno zachované políčko pomocou troch čísel x_i, y_i a c_i ($1 \leq x_i, y_i \leq n$, $0 \leq c_i \leq 1$) – súradnice tohto políčka a jeho hodnota. Je zaručené, že jedno políčko sa môže objaviť na najviac jednom riadku vstupu.

Testovacie vstupy pre túto úlohu sú rozdelené do 8 sád, pre ktoré platia nasledovné dodatočné obmedzenia:

- V sade 1 platí, že $n = m$ a všetky zadané políčka ležia na hlavnej diagonále. Inými slovami, máte presne zadané ako vyzerá hlavná diagonála a nič navyše.
- V sade 2 a 3 platí, že $n \leq m$ a medzi zadanými políčkami sa vyskytujú všetky políčka na hlavnej diagonále. Oproti vstupom zo sady 1 teda máte aj zadaných aj niekoľko dodatočných políčok.
- V sade 4 platí, že $n \leq 20$.
- V zvyšných sádach neplatia žiadne dodatočné obmedzenia.

Formát výstupu

Na výstup vypíšte jedno celé číslo – počet možností, ktorými je možné doplniť maticu $n \times n$ číslami 0 a 1 tak, aby platila zadaná podmienka. Toto číslo vypíšte modulo 1 000 000 009.

Príklad

vstup	výstup
<pre>3 3 1 3 1 3 2 0 1 1 1</pre>	8

Všetky možné doplnenia sú uvedené na nasledovnom obrázku:

<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	1	0	0	<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	1	0	<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr></table>	1	0	0	0	1	0	1	0	1	<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	1	1	0	1	1	0	1	1	0
1	1	1																																					
0	0	0																																					
1	0	0																																					
1	0	0																																					
0	1	0																																					
1	1	0																																					
1	0	0																																					
0	1	0																																					
1	0	1																																					
1	1	0																																					
1	1	0																																					
1	1	0																																					
<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr></table>	1	1	0	1	1	0	1	0	1	<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr></table>	1	0	1	1	0	0	1	0	0	<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	0	0	0	1	1	1	<table border="1" style="border-collapse: collapse;"><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	0	1	1	0	0	1	1	1
1	1	0																																					
1	1	0																																					
1	0	1																																					
1	0	1																																					
1	0	0																																					
1	0	0																																					
1	1	1																																					
0	0	0																																					
1	1	1																																					
1	0	1																																					
1	0	0																																					
1	1	1																																					

vstup	výstup
<pre>3 6 1 1 1 1 2 1 2 1 1 3 3 0 3 2 0 2 3 0</pre>	0

Ak do stredného políčka dopíšeme 0, štvorec 2×2 v ľavom hornom rohu nebude mať párne veľa jednotiek. Ak do stredného políčka dopíšeme 1, zlý bude pravý dolný 2×2 štvorec.

7. Ye ti zima?

12 b za popis, 8 b za program

Yeti Ignác nedávno dostal skvelý nápad: oholí si nohy, aby bol krajší. No len čo tak spravil, pochopil, že ten nápad nebol až taký skvelý. Keď sa teraz brodí snehom, je mu zima. Rozhodol sa preto, že si pustí na počítači vaše programy a tým sa zahreje.

Vytrestajte Ignáca za jeho hlúpe nápady, nech si dobre zapamätá, že yeti si nemá holiť nohy. Vyriešte teda čo najefektívnejšie nasledujúcu úlohu, nech sa yeti príliš nezahreje.

Úloha

KSP-krása reťazca je počet spôsobov, ktorými v ňom vieme vyznačiť podpostupnosť **ksp** – teda niekde vyznačiť písmeno **k**, niekde napravo od neho písmeno **s** a napravo od toho zase písmeno **p**. Napríklad *KSP-krása* reťazca **kokosplesk** je 2: buď zoberieme znaky na indexoch 0, 4, 5 alebo znaky na indexoch 2, 4, 5.

Na vstupe dostanete jeden reťazec S a potom postupne q otázok o ňom. Každá otázka bude nasledujúceho tvaru: “Aká je *KSP-krása* podreťazca tvoreného znakmi na indexoch a_i až b_i ?”

Formát vstupu

V prvom riadku je číslo n : dĺžka reťazca. V druhom riadku je reťazec S tvorený n malými písmenami anglickej abecedy. V treťom riadku je číslo q : počet otázok.

Zvyšok vstupu tvorí q riadkov. Tieto riadky a aj im zodpovedajúce otázky si očisľujeme od 0 po $q - 1$. Správnu odpoveď na otázku i označíme c_i a navyše dodefinujeme, že $c_{(-1)} = 0$.

Riadok popisujúci otázku číslo i obsahuje dve celé čísla x_i a y_i , pre ktoré platí $0 \leq x_i, y_i < n$. Hodnoty a_i a b_i pre túto otázku sú určené nasledovne: Nech $p_i = (x_i + c_{i-1}) \bmod n$ a $q_i = (y_i + c_{i-1}) \bmod n$. Potom $a_i = \min(p_i, q_i)$ a $b_i = \max(p_i, q_i)$.

V jednotlivých sadách testov platia pre n a q nasledovné obmedzenia zhora:

Sada	1	2	3	4
n, q	50	20 000	100 000	500 000

Formát výstupu

Vypíšte q riadkov a v nich postupne čísla c_0 až c_{q-1} .

Upozornenie

Pri hodnotení popisov budeme o čosi viac ako inde prihliadať na asymptotickú časovú zložitosť vašich riešení. Špeciálne upozorňujeme na to, že algoritmy asymptoticky horšie od vzorového riešenia nedostanú plný počet bodov za popis (hoci môžu úspešne vyriešiť všetky testy).

Príklad

vstup	výstup
16 kksspkkokosplesk 4 6 15 4 14 3 3 15 0	2 8 0 16

- Otázka 0 sa pýta na podreťazec **kokosplesk**, ktorého *KSP-krása* je $c_0 = 2$.
- Keď vieme, že $c_0 = 2$, vypočítame si, že otázka 1 má $a_1 = 0$ a $b_1 = 6$, pýta sa teda na reťazec **kksspkk**. Toto *KSP-krása* je $c_1 = 8$.
- Otázka 2 sa pýta na nejaký jednoznakový reťazec, toho *KSP-krása* je určite $c_2 = 0$.
- No a na záver, otázka 3 sa teda pýta na celý reťazec S .

8. Pochúťka

12 b za popis, 8 b za program

Paulinka⁹ mala jedného dňa robiť úlohu z databáz. Ako písala riešenie, uvedomila si, že jedáleň práve zatvorila. Bola nedeľa, takže aj všetky obchody boli zatvorené. Žiaľ, musela pokračovať v písaní o hlade a tak miesto databáz Paulinka premýšľala o koláčoch. A tak to aj skončilo. Jej domáca úloha obsahovala databázy všelijakých koláčov.

Keď prišla z ďalekého západu na prázdniny domov, rozhodla sa, že taká databáza koláčov je výborný nápad. Išla ho teda zrealizovať. Po chvíli plánovania prišla na to, že databáza predsalen nie je najlepší spôsob pre uloženie koláčov a namiesto databázy upeče maticu koláčov.

⁹s krátkym i

Matica mala r riadkov a s stĺpcov a Paulinka upiekla $n + m$ koláčov. Najskôr prvých n položila tak, že i -ty bol v $(i \bmod r)$ -tom riadku a $(i \bmod s)$ -tom stĺpci.

Následne ešte na niektoré políčka matice položila zvyšných m koláčov tak, ako sa jej to páčilo. Paulinka mohla položiť aj viacero koláčov do jedného políčka.

Ako finalizovala svoju maticu, rozhodla sa, že niektoré koláče ozdobi maslovým krémom. Samozrejme, nie len tak ledabolo! Každý riadok a každý stĺpec musí obsahovať nepárny počet ozdobených koláčov. Samozrejme, žiadne rearanžovanie koláčov!

Paulinka s krémom stojí nad maticou a premýšľa, ako to urobiť. Ide to vôbec? Kolkými spôsobmi?

Úloha

Kolkými spôsobmi ide ozdobiť koláče v horeuvedenom rozložení tak, aby v každom riadku a stĺpci matice bol nepárny počet ozdobených koláčov?

Dva spôsoby sú rôzne, ak je nejaký koláč ozdobený v jednom a neozdobený v druhom. V jednom políčku môže byť aj viac koláčov.

Formát vstupu

Na prvom riadku dostanete rozmery mriežky $r, s \leq 10^9$ a čísla $r + s \leq n \leq 10^9$ a $m \leq 10^5$ oddelené medzerou.

Na ďalšom riadku je m čísel, r_0 až r_{r-1} , kde r_i je riadok $n + i$ -teho koláča.

Na treťom riadku je m čísel, s_0 až s_{s-1} , kde s_i je stĺpec $n + i$ -teho koláča.

Riadky a stĺpce číslujeme od nuly.

V jednotlivých sadách vstupov platia pre r, s, n a m nasledovné obmedzenia:

1	2	3	4	5	6	7	8
$r, s \leq 10$	$r \leq 10$	$r, s \leq 10^6$	$r, s \leq 10^6$	$r, s \leq 10^6$	$\gcd(r, s) \leq 10^6$	$r, s \leq 10^9$	$r, s \leq 10^9$
$n + m \leq 20$	$s < 10^3$	$n \leq 5(r + s)$	$n = r \cdot s$	$n \leq 5(r + s)$	$n \leq 10^9$	$n \leq 10^9$	$n \leq 10^9$
	$n \leq 2^{16}$	$m \leq 10^5$	$m = 0$	$m \leq 10^5$	$m \leq 10^5$	$m \leq 10^5$	$m \leq 10^5$
	$m \leq 10^5$						

Navyše, v tretej a siedmej sade stačí vypísať, či existuje aspoň jedno riešenie. Presnejšie, ak je odpoveď nula, tak program má vypísať nulu, a inak môže vypísať akékoľvek kladné číslo.

Formát výstupu

Vypíšte jedno číslo – počet rôznych spôsobov ozdobení koláčov. Keďže toto číslo môže byť veľmi veľké, vypíšte ho modulo 1 000 000 009.

Príklad

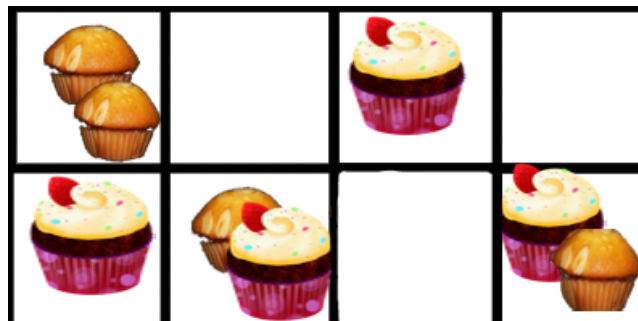
vstup

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

výstup

```
8
```

Jeden možný spôsob ozdobenia koláčov:



vstup

```
6 4 50 5
0 1 2 3 4
3 3 3 3 3
```

výstup

```
743544352
```

Tu je možností veľa.

vstup

2 4 6 2
1 0
3 2

výstup

0
