Korešpondenčný seminár z programovania
XXXIV. ročník, 2016/17
Katedra základov a vyučovania informatiky FMFI UK,
Mlynská Dolina, 842 48 Bratislava

Úlohy 1. kola letnej časti, kategória T

Termín odoslania riešení tejto série je pondelok **5. júna 2017**.

1. Trikrát bola párty

kat. T; 0 b za popis, 20 b za program

Tina rada organizuje párty pre svojich spolužiakov. Zorganizovať dobrú párty však nie je len tak. Medzi jej spolužiakmi panujú rôzne vzťahy. Napríklad Miro teraz strašne letí na Zuzu. Ak Tina pozve na párty Zuzu, určite musí pozvať aj Mira, inak by jej to do smrti vyčítal. No a Zuza sa minule strašne pohádala s Helou, a tak ak Tina pozve jednu z nich, určite nechce pozvať zároveň aj tú druhú. A tak ďalej.

Keď si tak Tina spísala všetky požiadavky koho môže a nemôže pozvať, zistila, že existujú **presne tri** možnosti, akú množinu spolužiakov na párty pozvať. Zorganizovala preto postupne tri párty – jednu pre každú možnosť.

Tina ti dala pre každú párty zoznam ľudí, ktorí sa jej zúčastnili. Nájdi jednu možnosť toho, ako vyzerajú vzťahy medzi ľuďmi v triede.

Úloha

Ľudí v triede si očíslujeme od 1 po n.

Budeme uvažovať len jeden typ podmienok: implikácie. Každá podmienka má teda tvar "ak (niekto je alebo nie je na párty) tak (niekto musí alebo nesmie byť na párty)". To, že človek x je/musí byť na párty, v podmienke zapíšeme $+x$. To, že tam nie je/nesmie byť, zapíšeme $-x$.

Na vstupe sú dané tri množiny ľudí, ktoré boli na párty. Zostrojte konkrétnu sadu implikácií, pre ktorú bude platiť, že existujú presne tri spôsoby, ako ich všetky naraz splniť, a tieto tri spôsoby zodpovedajú daným trom množinám ľudí.

Formát vstupu

V prvom riadku vstupu je číslo n: počet ľudí v triede. Zvyšok vstupu tvoria tri riadky. V každom riadku je n núl alebo jednotiek: postupne od človeka 1 po človeka n je tam 0 ak na párty nebol, a 1 ak na nej bol.

Vstupom môžu byť ľubovoľné tri n-tice núl a jednotiek. Nie je teda zaručené, že hľadaná sada implikácií musí existovať.

Je päť sád vstupov. Postupne majú maximálnu hodnotu n rovnú 3, 5, 10, 20 a 50.

Formát výstupu

Ak neexistuje žiadna sada implikácií s požadovanou vlastnosťou, vypíšte jeden riadok a v ňom číslo -1.

Ak existujú nejaké vyhovujúce sady implikácií, nájdite nejakú dostatočne malú. Presnejšie, sada, ktorú vypíšete, musí mať nanajvýš 470 implikácií. Vypíšte ju v nasledovnom formáte: najskôr riadok obsahujúci číslo p udávajúce počet implikácií, potom p riadkov a v každom z nich implikácia tvaru "`ak [cislo] tak [cislo]`".

(Je zaručené, že ak existuje nejaká sada implikácií, existuje aj dostatočne malá sada.)

Príklady

vstup

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

výstup

```
ak +1 tak -2
ak +3 tak +1
ak +3 tak +2
```

Tina má troch kamarátov. Nazvime ich Anka (1), Boris (2) a Cecil (3). Na prvej párty nebol nik z nich. Na druhej bol len Boris. Na tretej bola len Anka. Ako môže vyzerať sada implikácií, pre ktorú sú toto jediné tri možnosti, koho pozvať?

Existuje veľa možností. Tá uvedená v príklade výstupu má nasledujúci význam: Ak Tina pozve Anku, nesmie pozvať Borisa. Ak pozve Cecila, musí pozvať Anku aj Borisa.

Logicky si odvoďme, koho môže Tina pozvať pre túto sadu implikácií. Ak by pozvala Cecila, tak z druhej a tretej implikácie musí pozvať aj zvyšných dvoch, to však spôsobí spor s prvou implikáciou. Preto Cecil má smolu a na žiadnej párty nebude. Do úvahy tiež nepripadá párty, na ktorej sú Anka s Borisom – opäť pre spor s prvou implikáciou. No a ľahko overíme, že už ostali len tie tri možnosti, ktoré boli zadané na vstupe, a že žiadna z nich k sporu nevedie.

Existujú aj iné správne riešenia. Mohli sme napríklad použiť aj implikáciu "`ak +3 tak -3`*", z ktorej by tiež vyplývalo, že Cecil na žiadnu párty nesmie.*

vstup

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

výstup

```
-1
```

Neexistuje sada implikácií, pre ktorú by boli prípustné len tieto tri zloženia účastníkov párty a žiadne iné.

2. Totálny zabiják

kat. T; 0 b za popis, 20 b za program

Nebolo to tak dávno, čo bol náš Emko totálny šialený zabiják. Kešďe bola doba ťažká, jediný zamestnávateľ ochotný zamestnať ho bola *Korporátna spoločnosť pseudomafiánov.* Tí mali n nepriateľov, ktorých sa potrebovali zbaviť. Emko ale nebol žiadny lacný chlap, a tak si za odstránenie každého z nepriateľov účtoval nie nutne rôzne sumy.

Spoločnosť má momentálne obmedzené financie, a tak vie zaplatiť odstránenie len niekoľkých nepriateľov. Pre spoločnosť majú nepriatelia iné hodnoty, ako pre Emka, a ešte neživoria, preto nie je najlacnejšie riešenie nutne najlepšie. Potrebovali by teda vedieť k najmenších účtov, aké vie Emko vystaviť. Potom už ľahko vyberú najlepšiu variantu pre získanie vlády nad svetom.

Keďže ale Emko nepatrí ku najchytrejším a na pohovore mu namerali IQ rovné celých 26, je občas ochotný spoločnosti **zaplatiť** za objednávku. Teda spoločnosť vie dokonca okrem získania vlády občas aj zbohatnúť. Podieľajte sa na tejto akcii a buďte pri koryte aj VY. Pomôžte spoločnosti zistiť k najmenších účtov, aké nám vie Emko dať.

Úloha

Máte zadaných n čísiel: ceny, za ktoré je Emko ochotný odstrániť jednotlivých nepriateľov. Zistite hodnoty k najmenších **neprázdnych** rôznych účtov, ktoré môže Emko vystaviť. Každý účet je určený zoznamom nepriateľov, ktorých chceme odstrániť, a jeho hodnota je súčet cien týchto nepriateľov. Účty sú rôzne, ak zoznam ľudí na nich nie je identický.

Formát vstupu

Na prvom riadku vstupu sa nachádzajú dve čísla oddelené medzerou, a to $1 \leq n \leq 200\,000$ a $1 \leq k \leq \min(2^n - 1, 200\,000)$. Na druhom riadku bude n medzerou oddelených čísel a_i ($|a_i| \leq 10^9$), a to je cena, za ktorú je Emko ochotný odstrániť i-teho nepriateľa.

Sú štyri sady vstupov, s takýmito obmedzeniami:

číslo sady	1	2	3	4
$n \leq$	10	100	1 000	200 000
$k \leq$	1 023	200 000	200 000	200 000

Formát výstupu

Na výstup vypíšte k riadkov, a to hodnoty k najmenších možných účtov usporiadaných od najmenšieho. Na každom riadku vypíšte jednu hodnotu.

Príklady

vstup	výstup
2 3 -1 1	-1 0 1

vstup	výstup
3 7 -1 0 1	-1 -1 0 0 0 1 1

3. Triviálna interaktívna úloha

kat. T; 0 b za popis, 20 b za program

Na pondelňajší krúžok matematiky nikto okrem vás a učiteľky neprišiel, lebo všetci sú na lyžiarskom zájazde. Učiteľka sa preto rozhodla hodinu spraviť trochu interaktívnejšou, nakoľko sa nemusí venovať desiatim študentom naraz.

Úloha

Na začiatku si učiteľka tajne zvolí dve celé čísla: *modulus* m $(1 \leq m)$ a *stav* x_0 $(0 \leq x_0 < m)$. Vašou úlohou je tieto dve čísla uhádnuť pomocou niekoľkých otázok. Učiteľka vám na začiatku oznámi, že modulus je najviac n, a že sa môžete spýtať najviac q otázok.

Každá otázka vyzerá nasledovne: Vy si zvolíte jedno celé číslo a $(1 \leq a \leq n)$. Učiteľka zoberie aktuálny stav x_i a vypočíta z neho nový stav x_{i+1} použitím nasledovného vzorca: $x_{i+1} = (x_i + a) \bmod m$. Vám však nič z toho neukáže. Jediné, čo vám na konci oznámi, je, či je nový stav menší, rovnaký, alebo väčší ako predchádzajúci stav.

Kedykoľvek môžete skúsiť uhádnuť m a x_0, najneskôr ale po q otázkach. Môžete hádať iba raz.

Formát vstupu

Na začiatku dostanete celé číslo t: koľkokrát sa s vami bude učiteľka hrať (ak ju v predchádzajúcich hrách nesklamete).

Na začiatku každej hry dostanete celé čísla n a q. Po každej vašej otázke dostanete na samostatný riadok odpoveď: $>$ ak sa stav zväčšil, `=` ak je rovnaký, a $<$ ak sa zmenšil.

Formát výstupu

Každú otázku píšte do samostatného riadku, a musí pozostávať z jedného celého čísla a $(1 \leq a \leq n)$. Váš tip tiež vypíšte do samostatného riadku v tvare `0 m x0`.

Obmedzenia

Je päť testovacích sád, s nasledujúcimi obmedzeniami:

Sada	1	2	3	4	5
t	2 016	1 000	200	100	200
n	63	250	10 000	10^9	10^{18}
q	126	251	2 300	2 600	1 200

Poznámky

Nezabudnite po každej vašej otázke alebo tipe `flush`-núť výstup, aby sa hneď odoslal testovaču. (`cout.flush()` v C++, `fflush(stdio)` v C, `sys.stdout.flush()` pre Python, a `System.out.flush()` v Jave.)

Keď váš program zo vstupu prečíta `EOF` (čiže keď sa mu nepodarí načítať nejaký údaj), váš program by mal

hneď skončiť. Toto môže nastať napríklad vtedy, keď ste v práve skončenej hre zle odpovedali. Teraz čakáte ďalšie n a q, na vstupe vám však už žiadne údaje neprídu.

Ak bude váš program naďalej bežať a pokúšať sa komunikovať, môže sa stať, že namiesto `WA` dostanete `EXC`, `TLE`, alebo iné zveriny.

Príklady

vstup

```
2
2 9

<

>

<

1 9
```

výstup

```

1

1

1

0 2 1

0 1 0
```

4. Turnaj Veľkých Šachistov

kat. T; 0 b za popis, 20 b za program

V Slovakistane sa pred rokom rozhodli osadníci usporiadať národný šachový turnaj. Za prvé miesto sa dal získať diplom, a tak všetci hrali ako o život. Keďže sa však nehralo na čas, vyskytol sa problém – hráči nechceli priznať prehru. Stále len tvrdili, že určite nemajú mat, však určite sa z tej situácie dá nejak dostať, keby len mali ešte chvíľku na rozmýšlanie… A možno ešte jednu… Osadníci si teda na pomoc zavolali šikovných programátorov, ktorí ich problém vyriešili.

Tento rok osadníci plánujú usporiadať národný šachový turnaj zas. Samozrejme, tohtoročný turnaj musí byť lepší, napínavejší, veľkolepejší – a hlavne – väčší. Preto sa tento rok nebude na turnaji hrať šach na obyčajných šachovniciach rozmerov 8×8, ale na šachovniciach $n \times n$, a to s f figúrkami.

S týmto je však problém – programy šikovných programátorov totiž na takýchto veľkých šachovniciach už nestíhajú rozhodnúť, či má niektorý z hráčov šach alebo mat. Osadníci Slovakistanu teda opäť potrebujú pomoc!

Úloha

Pre daný popis šachovnice rozlíšte, či má niektorý z hráčov šach alebo mat. Pritom berte do úvahy len obyčajné pohyby figúrok (komplikované ťahy ako *rošáda*, *en passant*, *pohyb pešiakom o dva políčka vpred* a *povýšenie pešiaka* sa v Slovakistane neakceptujú).

Ak má práve jeden z hráčov šach, zistite tiež, koľko má platných ťahov (po ktorom už jeho kráľ nebude ohrozený).

Formát vstupu

V prvom riadku je číslo $1 \leq t \leq 3000$, udávajúce počet šachovníc na vstupe. Nasleduje t popisov šachovníc.

Popis šachovnice začína jedným riadkom s číslami $2 \leq n, f \leq 100\,000$: dĺžka strany šachovnice a počet figúrok na nej. Nasleduje f riadkov, každý popisujúci jednu figúrku súradnicami políčka na ktorom stojí x_i a y_i, a jej typom z_i. Políčko v ľavom hornom rohu má súradnice $(1, 1)$ a v pravom dolnom rohu (n, n). Znak z_i je písmeno označujúce typ figúrky; Každé z_i je jedno z písmen `KQRBHP`, reprezentujúce v tomto poradí kráľa, kráľovnú, vežu, strelca, koňa, a pešiaka.

Bielemu hráčovi patrí horná strana šachovnice (menšie y_i) a jeho figúrky sú označené malými písmenami. Teda bieli pešiaci sa pohybujú v kladnom smere y_i. Čiernemu hráčovi patrí dolná strana šachovnice a jeho figúrky sú označené veľkými písmenami. Jeho pešiaci sa pohybujú v zápornom smere y_i. Za každým popisom šachovnice je jeden prázdny riadok.

Platí $1 \leq x_i, y_i \leq n$; žiadne dve figúrky nestoja na rovnakom políčku, a na každej šachovnici je práve jeden biely kráľ `k` a jeden čierny kráľ `K`. Nakoniec môžete predpokladať, že vstupy sú rozumné, teda v každom vstupe súčet $n, f \leq 200\,000$.

V prvej sade $n = 8$.

Formát výstupu

Pre každú šachovnicu vypíšte jeden riadok s jednou z nasledovných hlášok:

- “Neutralna situacia.”, ak žiaden z kráľov nie je ohrozený nepriateľskou figúrkou.
- “Nemozna situacia.”, ak sú obaja králi ohrození nepriateľskou figúrkou.
- “{farba} hrac ma sach. Ma x platnych tahov.”, kde {farba} je “Biely”, resp. “Cierny”, ak je kráľ tohto hráča v ohrození, ale existuje x platných ťahov niektorými jeho figúrkami takých, po ktorom už v ohrození nebude.
- “{farba} hrac ma mat.”, kde {farba} je “Biely”, resp. “Cierny”, ak je kráľ tohto hráča v ohrození, a neexistuje platný ťah niektorou z jeho figúrok taký, po ktorom už v ohrození nebude.

Príklady

vstup

```
3
8 4
5 1 k
4 3 H
4 5 h
5 7 K

8 4
5 2 h
3 3 k
6 4 H
4 5 K

8 6
1 1 k
4 1 H
4 2 H
2 3 H
3 3 H
7 6 K
```

výstup

```
Nemozna situacia.
Neutralna situacia.
Biely hrac ma mat.
```

vstup

```
4
8 7
1 1 k
4 1 H
4 2 H
2 3 H
3 3 H
7 6 K
2 7 r

8 9
1 1 K
7 1 R
8 1 r
1 2 R
3 2 h
8 2 r
1 5 r
2 5 r
6 6 k

8 6
1 1 k
8 2 R
4 4 B
2 5 R
7 6 K
3 7 q

8 9
3 2 P
4 2 P
5 2 P
3 3 P
4 3 k
5 3 P
4 4 P
5 4 p
4 5 K
```

výstup

```
Biely hrac ma sach. Ma 1 platnych tahov.
Cierny hrac ma mat.
Biely hrac ma sach. Ma 1 platnych tahov.
Cierny hrac ma sach. Ma 5 platnych tahov.
```

5. Tam, kde žijú algoritmy...

kat. T; 0 b za popis, 20 b za program

Bol raz jeden algoritmus, ktorého každodennou prácou bolo hľadanie najdlhšej cesty v strome s vrcholmi od 1 po n. Jeho metóda bola nasledovná:

1. Strom si zakoreníme za vrchol s najmenším číslom.
2. Pre každého syna koreňa nájdeme najdlhšiu cestu, ktorej jeden koniec je samotný syn, a druhý koniec je v jeho podstrome. To spravíme obyčajným prehľadávaním do šírky.
3. Vyberieme dvoch synov s najdlhšími cestami. Keď tieto dve cesty prepojíme cez koreň, dostaneme najdlhšiu cestu prechádzajúcu koreňom.
4. Ešte sme ale nebrali do úvahy cesty, ktoré koreňom vôbec neprechádzajú. Každá taká cesta ale leží celá v podstrome niektorého syna. Stačí preto rekurzívne vyriešiť tieto podstromy – zakoreníme ich, spočítame pre synov koreňa najdlhšie cesty, ...

Kolega programátor mu poradil, že ak by za koreň vždy zvolil **centroid**[1] (pod)stromu, tak by mal oveľa viac voľného času, lebo by vraj bežal v čase $O(n \log n)$ namiesto $O(n^2)$.

Algoritmus sa nad tým zamyslel, a všimol si, že jeden strom môže mať viac ako jeden centroid. V takých prípadoch by sa musel rozhodnúť, ktorý z centroidov vybrať. S tým sa mu ale nechce otravovať. Zaujíma ho preto, koľko stromov s n vrcholmi je takých, že práca na ňom podľa nového postupu bude čisto manuálna, a nebude musieť počas nej robiť žiadne rozhodnutia.

Úloha

Strom je *ľahký*, ak má práve jeden centroid, a jeho odobratím sa strom rozpadne na niekoľko ľahkých stromov. Pre zadaný počet vrcholov n zistite, koľko rôznych stromov s n vrcholmi je ľahkých. (Dva stromy považujeme za rôzne, ak existujú vrcholy u, v, ktoré sú hranou spojené v prvom strome, ale nie v druhom.) Toto číslo môže byť veľké, vypíšte preto jeho zvyšok po delení prvočíslom p.

Formát vstupu

Na jedinom riadku vstupu sú dve celé čísla n, p oddelené medzerou. n udáva počet vrcholov a p je prvočíslo, po delení ktorým berieme zvyšok.

Je päť testovacích sád. Vo všetkých platí $1 \leq n \leq 3\,000$, a $n < p \leq 10^9$. Pre jednotlivé sady máme nasledovné obmedzenia:

číslo sady	1	2	3	4	5
$n \leq$	9	100	500	1 000	3 000

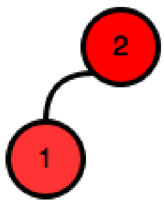
Formát výstupu

Na jediný riadok výstupu vypíšte jedno celé číslo z $\{0, 1, \ldots, p-1\}$ – počet vyhovujúcich stromov s n vrcholmi, modulo p.

Príklady

vstup	výstup
2 103	0

Pre $n = 2$ existuje iba jeden strom:

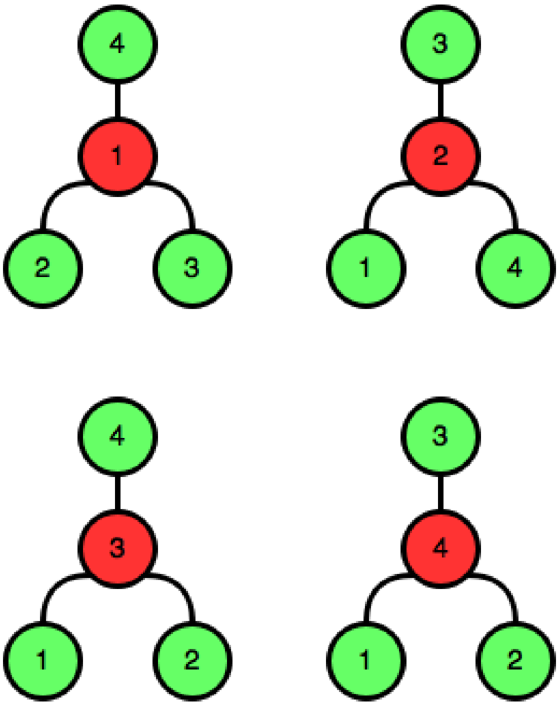


Tento strom nevyhovuje, nakoľko v prvom kroku môžeme za koreň zvoliť aj vrchol 1, aj 2.

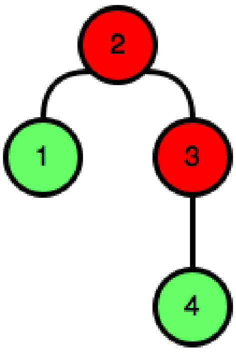
vstup	výstup
4 103	4

Pre $n = 4$ máme tieto štyri vyhovujúce stromy:

[1] taký vrchol v strome, ktorého každý syn má podstrom obsahujúci nanajvýš polovicu všetkých vrcholov



Príklad nevyhovujúceho stromu je



nakoľko môžeme v prvom kroku zvoliť za koreň 2 *aj* 3*.*