

fiks!



ČESKÉ
VYSOKÉ
UČENÍ
TECHNICKÉ
V PRAZE

FIT

Fitácký Informatický Korespondenční Seminář

Ročník 2015/2016, 1. kolo

Co je to FIKS?

FIKS je Fitácký Informatický Korespondenční Seminář pro středoškolské studenty pořádaný Fakultou informačních technologií ČVUT v Praze. Byl založen na podzim roku 2013 a nyní tak probíhá druhý ročník (samozřejmě číslujeme od nuly). Nabízí možnost potrápít tvůj mozek řešením algoritmických úloh různé obtížnosti, od snadných po zapeklité, na nichž se můžeš leccos nového naučit a podstatně se zdokonalit.

Jak to probíhá?

Jeden ročník se skládá z několika kol a následného soustředění pro nejlepší řešitele. V těchto kolech, která trvají vždy přibližně dva měsíce, máš možnost v teple domova řešit zadané úlohy, a své řešení nám potom odešleš. My ti toto řešení opravíme, obodujeme a pošleme zpět, aby ses mohl poučit ze svých chyb. Spolu s tím zveřejníme vzorové řešení, které můžeš prostudovat a třeba se něco přiučit. Získané body se sčítají do konečného žebříčku, ze kterého vybereme ty nejlepší a pozveme je na již zmíněné soustředění.

Proč řešit FIKS?

Řešením každého problému, se kterým se potýkáme, se zdokonalujeme. Zde ti nabízíme možnost pořádně se zamyslet nad zajímavými algoritmickými problémy, vyzkoušet své algoritmické myšlení a programátorské dovednosti a naučit se spoustu nových věcí.

Také je to možnost seznámení s novými lidmi, které baví informatika, programování, matematika a přemýšlení vůbec. Těm nejlepším jsme schopni garantovat přijetí na FIT ČVUT bez přijímacích zkoušek.

Jak se můžu zapojit?

Začni nejprve tím, že se zaregistruješ na našich webových stránkách na adrese <http://fiks.fit.cvut.cz>. Potom si stáhni zadání úloh (nebo využij tuto brožurku), vyřeš je a své řešení nám tamtéž odevzdej.

Typy úloh

Celkem se ve FIKSu můžeš setkat se třemi typy úloh. O který typ úlohy se jedná, je vždy uvedeno u konkrétního zadání úlohy.

Nejčastěji se u nás potkáš s úlohami typu *Rozmysli, popiš a naprogramuj*. U každé úlohy tohoto typu se odevzdává jak popis algoritmu (s odhadem asymptotické složitosti), tak i zdrojový kód řešení problému v tebou zvoleném jazyce (jakýkoliv vyšší programovací jazyk dle tvé volby, například C, Java, Pascal, apod.).

Dalším typem jsou úlohy *Zamysli se*. Tyto úlohy jsou obvykle více teoretické a vyžadují, aby ses nad nimi důkladně zamyslel. Oproti předchozímu typu úloh nemusíš nic programovat, odevzdává se pouze slovní popis řešení problému.

Pokud nemáš rád teoretické úlohy a raději by sis procvičil/a své programátorské umění, pak pro je pro tebe určena kategorie *Odpověz Sfinze*. V úlohách tohoto typu po tobě nechceme popis algoritmu, je však potřeba vyřešit daný problém a toto řešení pak precizně naprogramovat. Oproti ostatním typům úloh se navíc okamžitě dozvíš, zda je tvé řešení správné, protože ho můžeš okamžitě odevzdat do našeho vyhodnocovacího systému.

Další a podrobnější informace nalezneš na našich webových stránkách.

Milý řešiteli FIKSu!

Jsou již za námi dva roky, které byly plné řešení zajímavých úloh, následného programování řešení a v neposlední řadě také navazování přátelství s novými kamarády. Jako organizátoři bychom tak chtěli poděkovat všem, kteří pravidelně v dvouměsíčních intervalech oprašují své klávesnice, aby vybojovali umístění mezi těmi nejlepšími řešiteli.

V rukou držíš brožurku obsahující zadání 1. kola letošního ročníku FIKSu. Pokud jsi nový řešitel, tak nezoufej, jelikož na vedlejší stránce si můžeš přečíst základní informace, které k řešení úloh budeš potřebovat. Pokud jsi již někdy FIKS řešil, tak věz, že v této brožurce naleznáš v kompaktní formě všechna zadání pro 1. kolo. Tak či onak ti nic nebrání vrhnout se do řešení úloh.

Úlohy budou letos, ostatně jako vždy, zvláště zapeklité. Letos však hlavně z toho důvodu, že se s FIKSem vrhneš do prozkoumávání vesmíru. V nulové gravitaci totiž nemůžeš počítat s tím, že ti budou myšlenky poslušně sedět v hlavě. Dokážeš se postavit této výzvě?

Tvá cesta tímto ročníkem začíná výpravou na nedalekou planetu Mars. Přejeme ti tak hodně zdaru nejen při překonávání nástrah, které na této rudé planetě čekají. . .

Tvoji organizátoři

Fitácký Informatický Korespondenční Seminář

Ročník 2015/2016, 1. kolo

Začátek kola: 24. 8. 2015 00:00

Termín odevzdání: 18. 10. 2015 23:59

Odevzdávání: Přes webové rozhraní na <http://fiks.fit.cvut.cz>

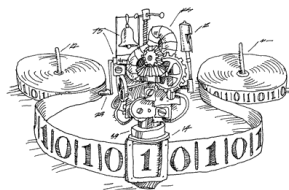
Další informace: <http://fiks.fit.cvut.cz>
kontakt@fiks.fit.cvut.cz



fiks!

Úloha č. 1

Lexikální analyzátor



Odpověz Sfinze!

10 b

Tato úloha je vyhodnocována automaticky. Je potřeba, aby výstup programu přesně korespondoval se specifikací výstupu níže. Jak odevzdávat tento typ úloh se můžeš dočíst na webových stránkách FIKSu pod záložkou „Jak řešit FIKS“.

Mise na Mars si žádá to nejmodernější vybavení a přesně takové je k nalezení na vesmírné lodi USS Flexington. Hlavním mozkiem lodi je počítač nejnovější generace schopný provádět i ty nejsložitější výpočty. Aby však počítač mohl něco zpracovávat, je potřeba mu nejprve nějakým způsobem problém zadat. Hlasové ovládání je zatím mimo provoz, inženýrská sekce na opravě pracuje. Kapitán si přeje provést simulaci přistání a k tomu je třeba ověřit základní logické operace. Kvůli opravám bude muset praporečik Paxson použít řádkový vstup z textové konzole na operačním stanovišti. Napsat program pro zpracování vstupu bude hračka.

Na vstupní řádce jsou celkem tři prvky. Paxson těmto prvkům říká *lexikální jednotky* nebo též *lexémy*. Každý lexém se může skládat z více znaků, praporečik Paxson tedy musí opatrně zjišťovat, kde končí jeden lexém a začíná jiný. Když jsou všechny lexémy úspěšně načteny, je potřeba jim dodat jejich syntaktický význam. První a třetí lexém jsou operandy, prostřední lexém představuje operátor. Když je i toto zpracované, nic už Paxsonovi nebrání v provedení simulace.

Vstup

Každý vstup na prvním řádku obsahuje celé číslo N . Dále následuje N řádků, každý z nich reprezentuje jedno zadání. Řádek se zadáním obsahuje jeden matematický výraz vyjadřující podmínku. Logický výraz je ve formátu $a \text{ OP } b$, kde a, b jsou celá čísla a OP je jeden z následujících operátorů: $<$, \leq , $=$, $!$, $=$, $>=$, $>$. Číslo a a operátor jsou vždy odděleny právě jednou mezerou. Maximální rozsah a, b a velikost N jsou $0 < N \leq 1000$ a $-2^{30} \leq a, b \leq 2^{30}$.

Výstup

Výstup obsahuje N řádků. Každý řádek odpovídá jednomu zadání. Obsahem každého řádku je řetězec **TRUE** nebo **FALSE** charakterizující logickou hodnotu, která odpovídá vyhodnocení podmínky.

Ukázkové vstupy

Vstup

```
2
1 <= 1
-1 != 1
```

Výstup

```
TRUE
TRUE
```

Vstup

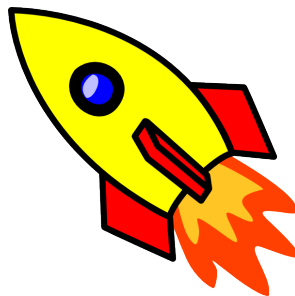
```
3
24 > 24
24 == 24
24 < 24
```

Výstup

```
FALSE
TRUE
FALSE
```

Úloha č. 2

Krizová situace



Odpověz Sfinze!

10 b

Tato úloha je vyhodnocována automaticky. Je potřeba, aby výstup programu přesně korespondoval se specifikací výstupu níže. Jak odevzdávat tento typ úloh se můžeš dočíst na webových stránkách FIKSu pod záložkou „Jak řešit FIKS“.

Náš křižník se pomalu blíží ke svému cíli, všechny systémy fungují na 100%. Posádka je shromážděna na můstku, kapitán se nadechuje k závěrečnému proslovu a v tom se rozezní siréna. **”POZOR, POZOR. Selhání brzdícího subsystému. Je nutná manuální aktivace brzdících trysek. Opakuji. Selhání brzdícího subsystému. Je nutná manuální...”**

Scotty, jakožto vrchní inženýr, se okamžitě zvedá a vybíhá do strojovny. Vzhledem k tomu, že strojovna je na nejnižším patře (v místnosti s číslem 1), kapitán se na vás obrací s následujícím úkolem. Určete, za jak dlouho se Scotty dostane do strojovny za předpokladu, že použije nejrychlejší možnou cestu.

Všechny místnosti jsou spojeny společnou chodbou, tj. z místnosti číslo a se dá za jednotku času dostat do místností číslo $a + 1$ a $a - 1$. Dále některé místnosti byly vybaveny jednosměrným teleportem umožňujícím se přemístit do některé z dalších místností (také za jednotku času). Konkrétně platí, že do každé místnosti s číslem $k \in \mathbb{N}$ se dá dostat pomocí teleportu z místností $2k$ a $3k$.

Vaším úkolem je tak určit minimální čas m , za který se dá dostat z můstku, tj. místnosti číslo n do místnosti 1.

Vstup

Vstup se skládá z několika testů. Každý test je na samostatném řádku a obsahuje přirozené číslo n , $1 \leq n \leq 10^8$, tj. číslo můstku.

Testovací soubor končí řádkem obsahujícím číslo 0. Tento test nemá být zpracováván.

Výstup

Na výstupu pro každý test bude jeden řádek obsahující jediné číslo m – minimální čas, za který je Scotty se schopen dostat z můstku do strojovny.

Ukázkové vstupy

Vstup

```
1
8
10
267
0
```

Výstup

```
0
3
3
7
```

Vstup

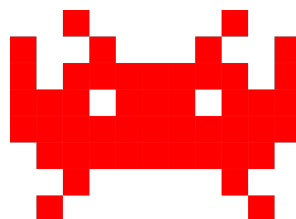
```
100000000
0
```

Výstup

```
25
```

Úloha č. 3

Útěk



Rozmysli, popiš a naprogramuj!

10 b

Kapitán vesmírné lodi narazil při cestě na Mars na letku nepřátelských marťanů a byť se snažil bojovat, byl nakonec i s posádkou zajat. Nicméně ještě není vše ztraceno. Marťané mají svoji oblíbenou hru, která poskytuje jejich obětem poslední šanci na přežití. Kapitán i s posádkou dostanou šanci si ji zahrát a uniknout tak jistě smrti.

Pravidla hry jsou následující – kapitán s posádkou dostanou plně vybavenou loď (ze které ovšem marťani z pochopitelných důvodů vymontovali všechny zbraně) a zadání formace nepřátelských lodí. Načež se po odstartování honu musí kapitán i se svou posádkou snažit těžce ozbrojeným marťanům utéct nedalekým intergalaktickým komínem. Aby pronásledování bylo zajímavé, dostane kapitánova loď náskok a dále platí, že všechny zúčastněné lodě mají stejný výkon motorů – hon tak mohou Marťané ukončit pouze sestřelením lodi kapitána (jelikož loď o shodném výkonu motorů zcela jistě nedohoní).

Protože se již po velmi dlouhou dobu nestalo, aby zajatí nešťastníci Marťanům unikli, obdržel kapitán ještě jednu možnost volby. A to možnost vybrat si, která z marťanských lodí bude během honu od jeho lodi nejdále. Tato volba se může zdát být nepodstatnou, nicméně náš kapitán ji velmi rád uvítal. A to z toho důvodu, že díky svým zkušenostem ví, že marťanské formace lodí mají svou slabinu. Ta tkví v nedokonalosti jejich komunikační sítě. Používají totiž zastaralou verzi sítě o názvu FI₁₀.

Sít FI₁₀ funguje na principu krátkých telekomunikačních spojení mezi jednotlivými páry lodí. Tato spojení nazveme přímými. Pokud se potřebují dorozumět dvě lodě mezi kterými neexistuje takovéto spojení, musejí své zprávy zasílat nepřímo přes ostatní lodě. Nepřímé spojení mezi loděmi x a y definujeme jako posloupnost lodí l_1, l_2, \dots, l_m , kde pro všechna i , $1 \leq i < m$ platí, že mezi l_i a l_{i+1} , x a l_1 , y a l_m existuje přímé telekomunikační spojení, a dále pro všechna i, j , $1 \leq i, j \leq m$, platí, že $l_i \neq l_j$. Tato posloupnost charakterizuje, jak se bude zpráva vyslaná z lodě x přeposílat až k lodi y . To, že se všechny lodě dorozumí, je zajištěno následujícími předpisy pro používání sítě FI₁₀:

- Přímé telekomunikační spojení mezi loděmi je vždy obousměrné a probíhá na vzdálenost právě 1 KiF,
- mezi libovolnými dvěma loděmi existuje právě jeden způsob, jak v rámci sítě komunikovat – tj. pokud je mezi danými dvěma loděmi x a y přímé spojení, tak pro ně neexistuje žádné nepřímé spojení, a pokud mezi loděmi x a y přímé spojení není, tak pro tuto dvojici existuje jediné nepřímé spojení (tj. jediná posloupnost lodí)

Je tak zřejmé, že rozhodnutí kapitána o tom, která z marťanských lodí bude během honu nejdálší, má velký vliv. Marťanské lodě totiž musejí po celou dobu honu striktně dodržovat formaci určenou komunikační sítí FI₁₀. Výběr nejdálší lodě formace od kapitánovy lodi tak přímo ovlivní i vzdálenost lodě, která bude ke kapitánově lodi nejbližší. Tuto vzdálenost chce kapitán samozřejmě maximalizovat, aby zmenšil šanci na zasáhnutí své lodě. Lze předpokládat, že se Marťané budou ke kapitánovi snažit přiblížit co nejvíce to půjde. Jinými slovy platí, že všechny marťanské lodě budou směřovat přímo ke kapitánově lodi bez zbytečných odboček, kvůli kterým by nevyužili optimálně délku přímých spojení v síti.

Hon vypukne za pár okamžiků a kapitán si stále není jist, kterou loď na základě formace marťanských lodí vybrat. Pomůžete mu v této nezáviděníhodné situaci?

Vstup

Na prvním řádku je zadáno číslo n , $0 < n \leq 10000$, které udává, kolik lodí tvoří nepřátelskou formaci. Dalších $n - 1$ řádků obsahuje dvojice lodí ve formátu a , b ,

kde $0 < a, b \leq n$. Každá z těchto dvojic značí, že mezi loděmi a a b existuje přímé telekomunikační spojení v rámci sítě FI_{10} .

Výstup

Na výstupu bude jediné číslo k udávající číslo lodě, jejíž výběr zajistí kapitánovi nejvyšší šanci na útěk (dle zadaných kritérií). Pokud je takových lodí více, je jedno, kterou kapitánovi poradíte.

Ukázkové vstupy

Vstup

3
1 2
2 3

Výstup

2

Vstup

5
1 5
2 5
5 3
5 4

Výstup

5

Úloha č. 4

Filmová sbírka



Rozmysli, popiš a naprogramuj!

10 b

Kapitán vesmírné lodi očekává důležitá data z průzkumné sondy na Marsu. Ví, kolik dat sonda pošle. Při pohledu na lodní disk však s hrůzou zjistil, že je téměř celý plný a data ze sondy se tam všechna nevejdou. Přišel na to, že 98.4 % disku zabírají filmy seržanta Bixiho. Kapitán si tedy zavolal Bixiho na můstek a nakázal mu smazat filmy tak, aby byl dostatek místa k uložení dat ze sondy.

Bixi má každý film ve své sbírce ohodnocen jednou až pěti hvězdičkami. Nejhorší filmy jsou ohodnoceny jednou hvězdičkou, naopak nejlepší filmy mají hvězdiček pět. Bixi by raději smazal všechny filmy, které mají méně než n hvězdiček, jen aby nemusel smazat film s n hvězdičkami. Rozhodl se tedy, že když už musí smazat film s n hvězdičkami, smaže i všechny filmy, které mají méně než n hvězdiček. Dále by rád smazal co nejméně filmů, tudíž má-li více filmů stejný počet hvězdiček, raději smaže jeden velký film, než aby mazal více malých.

Zjistěte, kolik filmů Bixi smaže, aby uvolnil alespoň tolik místa, kolik po něm požaduje kapitán.

Vstup

Na prvním řádku je zadáno číslo S , které udává, kolik místa je třeba uvolnit (v MB).

Na druhém řádku je zadáno číslo R (víte, že $R \leq 10^5$). Poté následuje R řádků, kde na i -tém řádku je zadána velikost m_i filmu číslo i (v MB) a jeho oblíbenost (počet hvězdiček) n_i , čísla jsou oddělena mezerou.

Pro velikost m_i platí $0 < m_i < 2^{31}$, pro velikost S platí $0 < S < 2^{31}$. Počet hvězdiček n_i nabývá hodnot z množiny $\{1, 2, 3, 4, 5\}$.

Výstup

Na výstupu bude jediné číslo udávající počet filmů, které musí Bixi smazat. V případě, že Bixi nemůže uvolnit dostatek místa ani kdyby smazal všechny své filmy, vypíše se věta *Nejde to, kapitáne*

Ukázkové vstupy

Vstup

```
125
6
35 5
10 4
10 4
110 4
31 4
150 5
```

Výstup

```
2
```

Vstup

```
87
3
10 2
25 4
5 1
```

Výstup

```
Nejde to, kapitáne
```

Vstup

142
8
10 1
40 2
20 1
66 2
35 1
2056 5
15 1
120 4

Výstup

5

Vstup

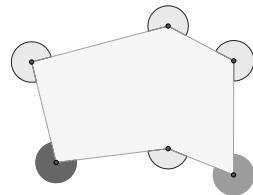
2042
7
10 1
10 2
10 3
10524 5
35 1
2056 4
15 1

Výstup

6

Úloha č. 5

Vedecké základne

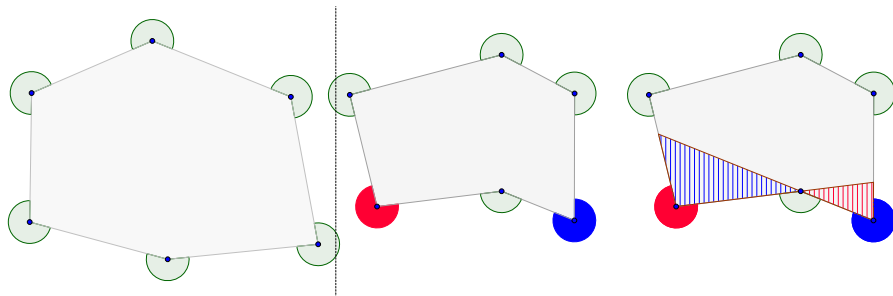


Rozmysli, popiš a naprogramuj!

10 b

Kapitán vesmírnej lodi plánuje vyslať na Mars tímy vedcov na dôkladný prieskum artefaktov, zaujímavých a podivných dát, ktoré vyslala sonda. Pre ich bezpečnosť im postaví dobre strážené základne. Zistil však, že má málo vycvičených vojakov. Tak sa musí použiť automatizované obranné veže a medzi nimi elektrický plot. Takže si dal navrhnuť rôzne základne v územiach na prieskum.

Obranné veže však automaticky likvidujú čokoľvek v stráženom úhľe. Ten je vymedzený plotom tak, že je definovaný stenami, ktorých sa veža dotýka, a je to vonkajší uhol. Z toho vyplýva, že pri nevhodnom návrhu základne môžu veže ohrozovať vnútro chránenej oblasti, a tak je nutné takéto návrhy vyhlásiť za nebezpečné (viď obrázok).



Vstup

Na prvom riadku bude celé číslo T , $1 \leq T \leq 1000$, udávajúce počet testovaných návrhov základní. Ďalej nasleduje T -krát popis návrhu základne. Každý takýto popis má na prvom riadku celé číslo N , $3 \leq N \leq 10^6$, čo je počet veží, ktoré popisovaný návrh obsahuje. Ďalej obsahuje N riadkov so súradnicami i -tej veže ($1 \leq i \leq N$) vo formáte $x_i y_i$, kde x_i a y_i sú celé čísla a platí, že $-10^6 \leq x_i, y_i \leq 10^6$.

Veže v rámci jedného opisu sú zoradené rovnakom v poradí, ako sa medzi nimi bude natahovať plot.

Výstup

Pre každý testovaný návrh základne vypíšte jeden riadok s reťazcom **bezpecny**, ak je návrh bezpečný, či **nebezpecny** v opačnom prípade.

Ukážkové vstupy

Vstup

```
2
6
1 2
3 1
5 2
5 4
3 5
1 4
6
1 2
3 3
5 2
4 7
3 10
1 5
```

Výstup

```
bezpecny
nebezpecny
```