

fiks!



ČESKÉ
VYSOKÉ
UČENÍ
TECHNICKÉ
V PRAZE

FIT

Fitácký Informatický Korespondenční Seminář

Ročník 2018/2019, 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 čtvrtý 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!

Další ročník FIKS je tu! Opět na tebe čekají tři kola po pěti úlohách, ve kterých si procvičíš dovednosti v řešení algoritmických úloh. Na rozdíl od minulých let budou jednotlivá kola delší a budou se z části překrývat.

V rukou držíš brožurku obsahující zadání 1. kola letošního ročníku FIKS. 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 naleznesh v kompaktní formě všechna zadání pro 1. kolo. Tak či onak ti nic nebrání vrhnout se do řešení úloh.

Letos se v úlohách setkáme s Eleanor, která žije ve válkou poznamenané společnosti. Efektivní řešení problémů je přísně zakázané, a lidé, kteří šíří myšlenky o efektivních algoritmech, jsou vyhledáni a umlčeni. Eleanor však vidí, že záměrným zpomalováním se marní mnoho času, a tak v duchu sní o rychlých algoritmech.

V následující části brožurky se dočteš příběh, který se bude v průběhu kol dále odkrývat. Poslední kapitola a celkové rozuzlení se bude odehrávat na jarním soustředění.

Tvoji organizátoři

Fitácký Informatický Korespondenční Seminář Ročník 2018/2019, 1. kolo

Začátek kola: 1. 10. 2018 12:00
Termín odevzdání: 31. 1. 2019 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!

Kapitola 0

Píše se rok 2056. Po skončení Velké války algoritmů (známá také jako Třetí světová kyber-válka) v roce 2039, je svět sjednocen pod jeden striktní režim, který potlačuje veškeré náznaky efektivních algoritmů, optimalizací a jiných fint (bitset). Je obecně známé, že tyto metody mohly za nadměrně rychlý vývoj technologií, zapříčinily závod ve zbrojení, a tak i vznik války. Používání efektivních algoritmů, či jejich pouhé psaní, je počín trestaný smrtí. Režim funguje zejména díky svým úřadům pro potírání algoritmů. Nejdůležitější z nich je FIT, který se zabývá vyhledáváním zmínek o efektivních algoritmech, jejich následnou cenzuru a předávání viníků orgánům činným v potíracím řízení.

Hrstka lidí, kteří nahlíží na systém nelibně, se seskupila do odbojové organizace FIKS. Snaží se o zachování efektivních algoritmů a jejich velkým cílem je osvobodit lid od systému. Fiksáci (název pro členy odboje) jsou systémem vyhledávání a bez milosti likvidováni, tak musí operovat ve stínech, a není známé, kdo ho vede, ani jakou mají organizaci.

Eleanor Sechsterer se narodila během války v roce 2037 v Praze. V roce 2040 byli oba její rodiče zajati a následně odsouzeni režimem pro jejich působení ve Velké válce algoritmů (zejména psaní efektivních toků). Eleanor byla vychovávána jako sirotek v režimem spravované internátní škole. Byla problémové a velmi netrpělivé dítě. Často podvědomě vymýšlela algoritmy efektivnější než bylo nutné, ale naučila se svoje názory skrývat, aby se nedostala do problémů.

S jejími 18 narozeninami byla povolána na FIT, aby pracovala na oddělení cenzury. Pro její nadání se již po roce stala seniorní cenzorkou, ovšem ona svou práci nenávidí. Dnes je den jako každý jiný, ovšem jeho průběh navždy změní její život.

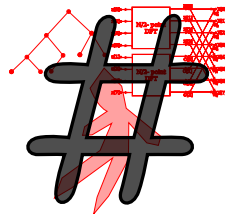
Kapitola 1

Jako každý pracovní den se Eleanor v sobotu ráno vydala do práce. Přišla do své kanceláře v oddělení cenzury FIT a prošla seznamem úkolů, které bylo každé ráno potřeba udělat. Seřadila příchozí dopisy pomocí opakovaného hledání toho, který je abecedně nejmenší. Vyškrtala černou cenzorofiksou nevhodné výrazy ze zítřejších novin a předala je ke zpracování editorům.

Po třech hodinách práce Eleanor dokončila tyto triviální úkoly. Frustrovaná banalitou ranních úkolů a způsobenou ztrátou času se Eleanor pustila do čtení úkolů, které cenzorky dostaly pro dnešek na vyřešení. Dnešní seznam byl obzvláště vypečený, a tak se pomalu smířovala s tím, že úkoly nestihnou vyřešit. „Přeci se jen tak nevzdám,“ řekla si. Dopad za nesplnění plánu by postihl zbytek cenzorek, a nejspíše by některé přišly o práci. Eleanor se tiše rozhodla, že udělá všechno pro to, aby úlohy vyřešila včas.

Úloha č. 1

Zaměstnanecké beneFITy



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“.*

Ve státní věznici pro efektivní programátory je stavěn nový systém podzemních chodeb. Z důvodu úspory peněz byl dozorcům odebrán benefit návštěvy FITness centra a jako náhrada mají posloužit nové pečlivě postavené chodby a přesné dávky jídla. Chodby je potřeba navrhnout tak, aby nejkratší cesta od vchodu k východu měla přesně určenou délku. Část věznice, ve které se chodby mají vytvořit si můžeme představit jako mřížku. Vchod se nachází v levém horním rohu a východ v pravém dolním rohu. Dozorci se umí pohybovat pouze na sousední políčka, kde sousední je takové, které s aktuálním políčkem sdílí jednu celou stranu.

Tvým úkolem je vyrobit pro každého dozorce plán chodeb.

Vstup

Na prvním řádku je jedno celé číslo $1 \leq t \leq 1000$ značící počet dozorců. Pak následuje t řádků. Každý z nich obsahuje 3 celá čísla m, n a k pro která platí $1 \leq m \leq 75, 1 \leq n \leq 75, n + m - 1 \leq k \leq 2 \left(\lfloor \frac{m-1}{4} \rfloor ((n-1) \bmod 4) + \lfloor \frac{n-1}{4} \rfloor (m-1) \right) + m + n - 1$. m a n jsou rozměry oblastí pro daného dozorce, k označuje přesnou délku nejkratší cesty od vchodu k východu (počet políček přes která musí dozorce projít včetně začátečního a koncového políčka).

Výstup

Pro každého z dozorců vypište m řádek po n znacích, kde každý znak je buď zed (#) nebo volné políčko (.). Pokud je více možných řešení, vypište libovolné z nich. Pokud nelze postavit chodbu podle daných rozměrů, vypište místo toho pro daného dozorce pouze jeden řádek s textem „Nejde to.“

Ukázkové vstupy

Vstup

```
3
7 15 37
12 7 45
2 16 21
```

Výstup

```
.....#..#.#
###.###.###.###
...#...#..#.#.
.#...#.#...##
.####.#.#.#.#.
.#.#...#.#.#.##
.....
Nejde to.
.#....#....#...#
...#.....#.....
```

Vstup

```
3
10 7 20
5 6 12
4 25 40
```

Výstup

```
..#...#
.#..#..
.#...###
...#.#
.#.....
#...###.
#.#...#
#.#.#.#
...##.##
..#.....
.....
...##..
.#....
.##.##
#.....
...#...#.#.#.....#..#.#
...#.#...#.#.##...#.#.##
#...#.#...##...#.#...#.#
.....#.#...#.#.#.#.....
```

Úloha č. 2

Páska



Rozmysli, popiš a naprogramuj!

10 b

Odbila dvanáctá hodina a Eleanor se vydala na oběd do zdejší nově otevřené kulinařské senzace zvané Technická menza. Jen co vykročila ze dveří FITu, okamžitě pochopila troubení, které poslední hodinu slyšela z okna. Na křižovatce se srazila dvě samořídící auta. „Kolika takovým nehodám se už mohlo zabránit, kdyby tak bylo povoleno používat simulované žíhání,“ povzdechla si Eleanor. Když se dostala blíž ke křižovatce, všimla si nešťastně vyhlížejícího strážníka s telefonem u ucha. „Šéfe, už nemám dost pásky, abych s ní obehnal celou tu hromadu šrotu,“ slyšela ho mumlat do moderního zařízení Aikon 0133. „Nesmysl,“ ozvala se Eleanor, prohlížející si zbytek pásky, který držel strážník v ruce, „pokud budete s páskou šetřit, určitě Vám vystačí.“ Tvým úkolem je zjistit, jak dlouhou pásku by strážník potřeboval, aby mu skutečně vystačila.

Vstup

Na prvním řádku bude celé číslo $2 \leq N \leq 10^6$ – počet objektů, tedy částí z automobilů, nacházejících se na křižovatce. Následuje N řádků, na každém celé číslo $3 \leq K \leq 100$ a dále $2K$ celých čísel $x_1, y_1, \dots, x_K, y_K$ reprezentujících souřadnice krajních bodů objektu v rovině křižovatky.

Výstup

Výstupem je jedno reálné číslo s přesností na dvě desetinná místa, reprezentující minimální délku pásky potřebnou k ohraničení všech objektů. Ohraničení musí tvořit jeden souvislý kus a žádné kousky objektů nesmí z ohraničení vyčnívat.

Ukázkové vstupy

Vstup

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

Výstup

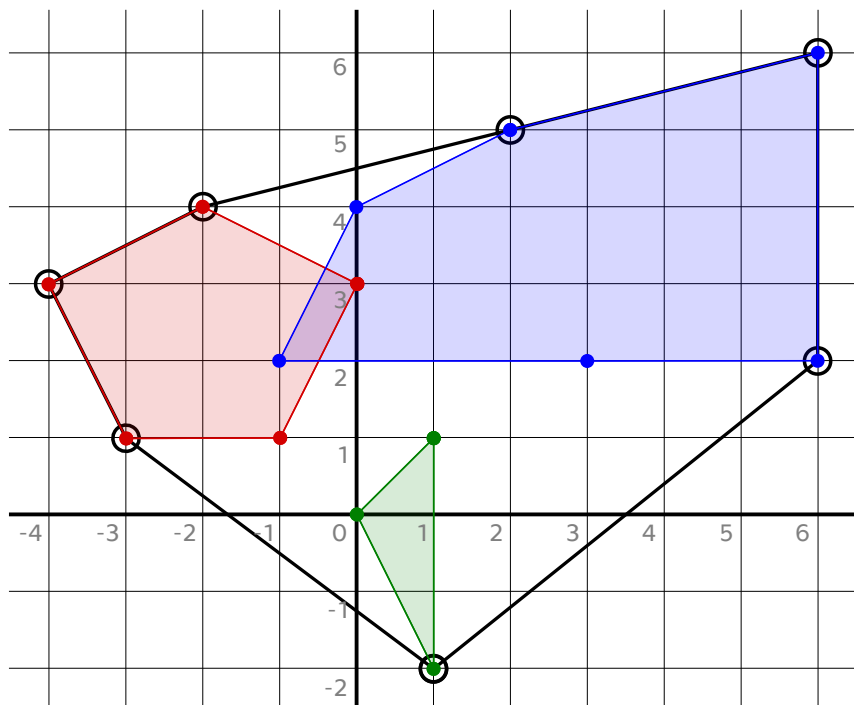
8.00

Vstup

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

Výstup

28.12



Obrázek 2.1 Znáornění druhého vstupu

Úloha č. 3

Nový kalendář



Rozmysli, popiš a naprogramuj!

10 b

Poté co se změnil režim, režim nezpochybnitelné pravdy. Režim proti efektivním algoritmům. Režim Velkého Vůdce. Vše, co uznávala předchozí vláda, je špatně. Musí to být nahrazeno něčím novým, **lepším**.

Jeden z cílů změny je i kalendář. Gregoriánský už není vyhovující, připomíná lidem až moc staré časy a je až moc efektivní. Je tedy nahrazen novým. Kalendářem Velkého Vůdce. Funguje následovně:

- Rok má 15 měsíců;
- 1., 6., 7., 9. a 15. měsíc má 25 dní;
- 2., 3., 8., 11. a 13. měsíc má 21 dní;
- 4., 5., 10., 12. a 14. měsíc má 24 dní;
- Poroz ale na to, že 3. měsíc je speciální. Každé tři roky je rok přestupný, ve kterém má 3. měsíc 22 dní. Výjimkou jsou ale roky dělitelné 100, to má 21 dní vždy, přestože je přestupný rok.
- Týden má 9 dní.

Začátek platnosti je stanoven na den narození Vůdce, tedy 20.8.1984. Tento den je nyní 1.1. rok 1 a jedná se o 1. den v týdnu. Tento rok není přestupným, prvním přestupným rokem bude rok 3.

Velký Vůdce je ale hodně náladový a svá rozhodnutí často mění. Jeden den platí tento kalendář, a druhý den platí zas úplně jiný. Eleanor už nestihá sledovat, co je vlastně za den. Tíží ji nutkání najít nějaký efektivní způsob, jak to zjistit. Ví akorát, co by bylo za den podle kalendáře Gregoriánského. Nezvádne to a nutkání brzy podlehne.

Je důležité, aby výsledný program byl srozumitelný, efektivní a kód se dal lehce přepsat při každé změně pravidel. Pomůžete ji s tím?

Vstup

Na vstupu je na prvním řádku jedno celé číslo $1 \leq N \leq 10000$ udávající počet dat z Gregoriánského kalendáře. Na každém z následujících N řádků jsou tři celá čísla d, m a r ($1 \leq d \leq 31$, $1 \leq m \leq 12$, $1984 \leq r \leq 100000$) oddělená mezerou. d značí den, m měsíc a r rok. Datum je zde ve formátu Gregoriánského kalendáře. Můžete počítat s tím, že datum na vstupu je vždy validní (tj. není třeba ošetřovat vstupy typu 31 2 2012).

Výstup

Na výstupu se očekává právě N řádků. Každý řádek na výstupu odpovídá svému řádku ze vstupu a jsou na něm právě 4 celá čísla dt , d , m a r znázorňující odpovídající den v týdnu, den, měsíc a rok v Kalendáři Velkého Vůdce.

Ukázkové vstupy

Vstup

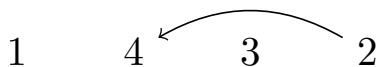
```
5
29 8 1984
30 8 1984
20 8 1985
6 12 3000
15 5 4000
```

Výstup

```
1 10 1 1
2 11 1 1
6 16 1 2
8 9 9 1060
6 4 9 2102
```

Úloha č. 4

Inverze



Zamysli se!

10 b

Tato úloha je čistě teoretická, tvým úkolem zde není napsat program. Namísto toho si dej záležet na kvalitním slovním popisu, kde mimo jiné jasně zdůvodníš, proč tvůj postup skutečně bude fungovat.

Jako každá vládní organizace i FIT musí dbát na bezpečnost svých dat. Před vstupem do interních informačních systémů se musí zaměstnanci autentizovat. Autentizace probíhá tak, že informační systém zadá uživateli 3 čísla N , K a R . Odpovědi uživatele by měla být posloupnost přirozených čísel $A := (a_1, a_2, \dots, a_N)$ délky N , ve které se smí vyskytovat maximálně K různých čísel a posloupnost A by měla obsahovat přesně R inverzí. Inverzí v posloupnosti čísel rozumíme takovou dvojici indexů $1 \leq i < j \leq N$, pro kterou platí, že $a_i > a_j$. Slovně řečeno, inverze v posloupnosti je dvojice čísel taková, že první je větší než druhé, a zároveň se první vyskytuje v posloupnosti dříve, než druhé.

Eleanor má za úkol pro každého zaměstnance vytvořit různou trojici N , K a R a příslušnou odpověď. K tomuto účelu FIT vyrobil počítačový program, který je ale příliš pomalý a Eleanor tráví většinu času čekáním nebo pracováním na jiných úkolech a občasným zapsáním výsledku. Konec dne se blíží, většina úloh je hotova, ale na autentizační kódy se stále čeká. Eleanor se nechce zůstat v práci až do noci, pomůžete jí vymyslet algoritmus, který úlohu vyřeší rychleji?

Vstup

Eleanor dostane 3 čísla N , K a R , která znamenají výslednou délku posloupnosti, počet povolených různých čísel a počet inverzí ve výsledku. Orientační meze hodnot jsou $1 \leq N \leq 10^6$, $1 \leq K \leq \infty$, $1 \leq R \leq \infty$ a orientační časový limit na vyřešení úlohy jsou 2 vteřiny.

Výstup

Odpovědí je buď libovolná posloupnost splňující popsané podmínky, nebo text „Neexistuje“ v případě, že takovou posloupnost nelze najít.

Ukázkové vstupy

Vstup

```
3 3 2
3 2 2
1 1 100
```

Výstup

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

Připomínáme, že úloha je teoretická a ukázkové vstupy, meze vstupních hodnot a časový limit slouží jen jako ukáзка. Řešením by měl být slovní popis algoritmu.

Úloha č. 5

Cenzoři



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“.*

Kvůli nadcházející inspekci byly všechny cenzorky z oddělení seřazeny do řady vedle sebe. Jelikož je cenzura pro stát velmi důležitá, cenzorek může být opravdu mnoho. Každá cenzorka má právě jedno unikátní ID. Budou kladeny dotazy, které se přímo dotýkají cenzorek na zadaném intervalu a kterým bude přidělena velmi citlivá práce. Tvým jediným, avšak velmi důležitým, úkolem bude na tyto otázky odpovídat. Pokud by se ti to nepodařilo, mohlo by to vést i k tvému uvěznění... a můžeš vsadit na to, že by se toto nepraví rozhodně nikdo nedozvěděl. S tvou reputací to ovšem nebude žádný problém, že?

Odpověď na dotaz se skládá ze tří částí:

- (i) Je důležité vědět jaké ID je na intervalu nejnižší, neb daná cenzorka má nejnižší postavení a tak pravděpodobně odedře veškerou práci.
- (ii) Je důležité vědět index (indexováno od 0) cenzorky, která má nejvyšší ID. Ta má totiž nejvyšší postavení a tak nese veškerou zodpovědnost.
- (iii) Je důležité vědět, jaký je *XOR* všech ID na segmentu. Jak jednou totiž řekl známý počítačový inženýr Georgy Xorwell, *XOR* je nejlepším hashem.

Vstup

První řádek vstupu se skládá z $1 \leq T \leq 100$, počtu testovacích sad.

Každá testovací sada pak začíná řádkem se dvěma čísly: $1 \leq N \leq 2 \cdot 10^5$ (počet cenzorek v řadě) a $1 \leq Q \leq 10^6$ (počet testovacích dotazů).

Další řádek pak obsahuje N čísel $0 \leq ID_i < N$, jednotlivá ID.

Dalších Q řádků pak obsahuje vždy 2 čísla $0 \leq B \leq E < N$, začátek dotazovaného intervalu a konec dotazovaného intervalu.

Výstup

Za každý dotaz odpověz na 3 řádkách vždy na každý typ:

- Na prvním řádku nejnižší ID na intervalu.

- Na druhém řádku index nejvyššího ID na intervalu.
- Na třetím řádku *XOR* všech ID na intervalu.

Pokud ti nepůjde některá z operací, stačí místo ní tisknout 0 (nebo něco podobného) a můžeš tak získat parciální body.

Ukázkové vstupy

Vstup	Výstup
3	2
4 1	1
2 3 1 0	1
0 1	0
6 2	2
5 1 0 3 4 2	0
2 2	1
0 1	0
3 2	4
1 2 0	0
0 2	1
1 2	3
	0
	1
	2