

Korešpondenčný seminár z programovania

Leták zimnej časti XXVIII. ročníka

Tento leták obsahuje zadania prvých kôl Korešpondenčného seminára z programovania kategórií KSP-Z, KSP-O aj KSP-T a pravidiel súťaže. Korešpondenčný seminár z programovania je súťaž v programovaní určená pre žiakov stredných škôl. Ročne sa jej zúčastňuje vyše 200 žiakov. Tento rok pre Vás skupina študentov a doktorandov FMFI UK pripravila už 28. ročník.

Organizácia:

Pre riešiteľov OI organizuje SK OI celoštátny korešpondenčný seminár ako súčasť OI. Zadanie úloh korešpondenčného seminára rozosiela riešiteľom niekoľkokrát do roka. Korešpondenčný seminár pre riešiteľov OI zo svojho kraja prípadne i z ďalších krajov resp. obvodov môžu organizovať aj KK OI. Pre najlepších riešiteľov týchto korešpondenčných seminárov sa obvykle taktiež organizujú sústreďenia.

Slovenský výbor Olympiády v informatike poveruje realizáciou Korešpondenčného seminára z programovania a s ním súvisiacich sústreďení občianske združenie **TROJSTEN**, ktoré zodpovedá za odbornú náplň a program seminára a sústreďení. Pri zabezpečovaní jednotlivých sústreďení bude OZ TROJSTEN spolupracovať s **IUVENTOU**, ktorá pokryje časť nákladov podľa schváleného rozpočtu OI.

Učiteľom informatiky:

Sprístupnite, prosím, tento materiál vašim žiakom. Ideálne by bolo spraviť niekoľko fotokópií zadaní a rozdať ich záujemcom, prípadne vyvesiť kópiu zadaní na nástenku. Podporte účasť študentov v našej súťaži. Môžu tak získať cenné vedomosti siahajúce nad rámec stredoškolských osnov. Možno práve na Vašej škole vyrastá budúci účastník Medzinárodnej informatickej olympiády. Mnohí z nich získavali skúsenosti práve účasťou v našom seminári.

Stredoškólakom:

Ak si žiakom strednej školy a vieš aspoň trochu programovať, neváhaj a zapoj sa do našej súťaže, má to množstvo výhod:

- Môžeš si porovnať svoje schopnosti so žiakmi z celého Slovenska.
- Riešením súťažných úloh a štúdiom poskytnutých vzorových riešení sa môžeš naučiť mnoho nových vecí. Získané poznatky a skúsenosti sa Ti iste budú hodiť na iných súťažiach v programovaní, pri prijímacích pohovoroch i počas vysokoškolského štúdia.
- Na riešenie úloh máš dosť času a môžeš ich riešiť doma bez toho, aby si niekam cestoval.
- Medzi zadaniami sa nachádzajú ľahšie aj ťažšie. Každý si môže vybrať tie, ktoré vie riešiť a ktoré sa mu zdajú zaujímavé.

- Najlepších riešiteľov pozývame každoročne na dve týždenné sústreďenia. Sústreďenie je jedinečná príležitosť ako spoznať nových priateľov s podobnými záujmami, naučiť sa veľa nového a zažiť veľa zábavy.

Rodičom a sponzorom:

Organizácia korešpondenčného seminára a sústreďení pre najlepších riešiteľov je z finančného hľadiska čím ďalej, tým náročnejšia. Preto s radosťou uvítame každého, kto by bol ochotný KSP sponzorovať. V prípade, že by ste KSP chceli sponzorovať a chcete presnejšie informácie, prípadne o nejakom sponzorovi viete, ozvite sa prosím mailom na ksp-info@ksp.sk.

.....

Priezvisko a meno: _____

Dátum narodenia: _____ Rok maturity: _____

Škola: _____

Adresa pre korešpondenciu:

Adresa domov:

E-mail: _____

Telefón: _____

28. ročník KSP – evidenčný lístok

.....

Priezvisko a meno: _____

Dátum narodenia: _____ Rok maturity: _____

Škola: _____

Adresa pre korešpondenciu:

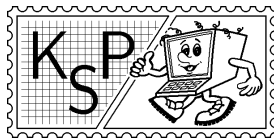
Adresa domov:

E-mail: _____

Telefón: _____

28. ročník KSP – evidenčný lístok

.....



Korešpondenčný seminár z programovania XXVIII. ročník, 2010/11

Katedra základov a vyučovania informatiky FMFI UK,
Mlynská Dolina, 842 48 Bratislava

KSP finančne podporujú: MICROSTEP-MIS spol. s r.o.

Agentúra na podporu výskumu a vývoja

Pravidlá

Korešpondenčný seminár z programovania (KSP) je súťaž programátorov – stredoškôľakov (a mladších). Jej cieľom je zdokonaľiť žiakov, ktorí už vedia programovať, alebo sa práve programovať učia, v programovaní a v algoritmickom myslení.

Kategórie: Korešpondenčný seminár z programovania pozostáva z troch kategórií: **KSP-Z**, **KSP-O** a **KSP-T**. Kategória KSP-Z je určená pre začínajúcich riešiteľov. Pokročilým riešiteľom je určená kategória KSP-O. Tým najlepším odporúčame riešiť **navyše** aj kategóriu KSP-T. Viac o kategórii KSP-T nájdete na konci pravidiel.

Kategórii KSP-O a KSP-T sa môžu zúčastniť všetci stredoškôľáci (a základôškôľáci) bez obmedzenia, aj riešitelia KSP-Z. Obzvlášť radi privítame účasť dievčat.

Kategóriu KSP-Z v tomto polroku môžu riešiť:

- druháci na strednej škole a mladší, ak do 1.9.2010 boli najviac na 1 sústreďení KSP
- tretiaci a štvrtáci na strednej škole, ak do 1.9.2010 neboli na sústreďení KSP

Za štvrtákov považujeme študentov, ktorí maturujú v tomto školskom roku, tretiaci sú tí, ktorí budú maturovať budúci školský rok, ostatné ročníky analogicky.

Odmena: Najlepších riešiteľov z kategórií KSP-Z a KSP-O pozývame každoročne na dve týždenné sústreďenia (pokiaľ sa ich podarí zorganizovať). Na jesenné sústreďenie nepozývame riešiteľov, ktorí už ukončili strednú školu.

Pokiaľ viete o niekom, kto by mohol sústreďenia KSP čo i len čiastočne sponzorovať, dajte nám o ňom (a prípadne jemu o nás) vedieť.

Organizácia súťaže: Všetky kategórie sú organizované v štyroch kolách, ktoré sú rozdelené na dva samostatné polroky po dvoch kolách. Na jarné sústreďenie teda budú pozývaní najlepší riešitelia podľa počtu bodov za zimnú časť a na jesenné sústreďenie podľa počtu bodov za letnú časť. Kolo obsahuje 10 príkladov, z čoho **príklady 1-5 tvoria kategóriu KSP-Z, príklady 4-8 kategóriu KSP-O a príklady 9-10 kategóriu KSP-T**. Príklady každého kola (nie nutne všetky) je treba vyriešiť a do určeného termínu odovzdať elektronicky (čo je nami preferovaný spôsob odovzdávania) alebo poslať na našu adresu. Riešenia odoslané po tomto termíne nemusia byť hodnotené, prípadne môžu byť riešiteľovi strhnuté body za neskoré odoslanie riešenia. Riešiteľom, ktorí sa zúčastnili niektorého kola súťaže, pošleme zadania ďalšieho kola, výsledkovú listinu a komentáre k riešeniam. Ďalšieho kola sa však môžu zúčastniť aj tí žiaci, ktorí sa predchádzajúceho kola nezúčastnili.

Požiadavky na riešenia: Základom riešenia každého príkladu (ak nie je v zadaní uvedené inak) je **listing programu** v ľubovoľnom vyššom programovacom jazyku (najlepšie Pascal, C/C++). Na programe nás zaujíma najmä jeho korektnosť (dáva program pre každý vstup správny výsledok?) a efektivita (koľko času a pamäte potrebuje na spracovanie vstupu určitej veľkosti). Dôraz kladte na algoritmickú časť programu, pri načítavaní vstupu a vypisovaní výstup nemusíte úplne presne dodržiavať formát zo zadania. Rovnako dôležitou súčasťou riešenia je **slovný popis** riešenia. Slovný popis by mal obsahovať popis použitého algoritmu, prípadne popis kľúčových dátových štruktúr. Mal by byť natoľko jasný a zrozumiteľný, aby bolo podľa neho možné napísať program rovnako efektívny, ako ten váš. Ďalej vyžadujeme

zdôvodnenie (dôkaz) **správnosti** použitého algoritmu a odhad časovej a pamäťovej zložitosti algoritmu (t.j. koľko času a pamäte potrebuje váš program v závislosti od veľkosti vstupných dát).

Na internetovej stránke <http://www.ksp.sk/ksp/riesenie.php> nájdete ukážkovo vyriešených niekoľko starších príkladov. Na tej istej stránke sa môžete aj dočítať, čo je vlastne časová a pamäťová zložitosť (ak vám tieto pojmy veľa nehovoria).

Technické detaily: Podobne ako v olympiáde z informatiky, ak píšete program v C++ a používate STL, súčasťou popisu vášho algoritmu by mal byť dostatočne podrobný popis implementácie dátových štruktúr, ktoré používate. Podobné podmienky platia aj pre používanie pokročilých dátových štruktúr v Jave či inom programovacom jazyku. Ak si nie ste istí, či niečo môžete použiť, radšej to popíšte, prípadne nás kontaktujte mailom na ksp-info@ksp.sk a opýtajte sa.

Spôsob doručenia: Použite formulár na adrese <http://www.ksp.sk/ksp2.0/eRiesenie>. Pokiaľ veľmi túžite odovzdávať riešenia na papieri, môžete použiť klasickú poštu.

Formálna úprava papierových riešení:

Riešenia píšete na papieri formátu A4. Prvý list každého riešenia má obsahovať hlavičku, v ktorej je meno riešiteľa, číslo príkladu, kategória, počet listov príkladu a škola. Ďalšie listy obsahujú meno, číslo príkladu a číslo listu. Ak má príklad niekoľko listov, zopnite ich spolu (v ľavom hornom rohu), aby sa nestratili. Nezopínajte listy patriace rôznym príkladom a už vôbec nepíšte riešenia viacerých príkladov na ten istý list papiera.

Formálna úprava elektronických riešení:

Akceptujeme riešenia len v textovom formáte alebo vo formáte pdf. Ak odovzdávate vo formáte pdf, zdrojový kód odovzdajte aj v textovom formáte. Každý príklad odovzdávajte v samostatnom súbore. Hlavička jednotlivých strán a formát strán je rovnaký ako pri papierových riešeníach.

Dôležité: Pred odovzdaním elektronického riešenia je potrebné sa zaregistrovať a vyplniť požadované kontaktné údaje. Odporúčame sa zaregistrovať pár dní pred dňom, kedy chcete odovzdať vaše riešenie pre prípad že by ste mali počas registrácie nejaké problémy. Riešenia sa odovzdávajú na nasledovnej stránke <http://ksp.sk/ksp2.0/eRiesenie>.

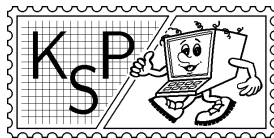
Bodovanie: Maximálny počet bodov za príklad závisí od jeho náročnosti, uvádzame ho pri zadaní každého príkladu. KSP je súťaž jednotlivcov, preto ak sa vyskytnú „kolektívne riešenia“, bude za ne udelených patrične menej bodov.

Evidenčný lístok: Každý súťažiaci, ktorý ešte KSP neriešil (alebo sa mu od minula zmenili niektoré údaje) musí vyplniť evidenčný lístok na adrese <http://www.ksp.sk/ksp2.0/ePrihlaska>. Veľmi dôležitou informáciou na tomto lístku je adresa pre korešpondenciu. Na túto adresu budeme posilať opravené riešenia a zadania ďalších kôl. Ak táto adresa je iná, ako vaša domáca adresa (ak bývate napr. na internáte), napíšte nám aj svoju domácu adresu, aby sme vám mohli písať aj cez prázdniny. Nezabudnite tiež uviesť rok, v ktorom maturujete.

Upozornenie: Účasťou v KSP nám zároveň dávate súhlas spracovať a archivovať v elektronickej podobe údaje, ktoré nám poskytnete na evidenčnom lístku, ako aj súhlas zverejniť vaše meno, školu, ročník a dosiahnuté výsledky vo výsledkovej listine.

KSP-T: Úlohou tejto kategórie je hlavne celoročná príprava riešiteľov na medzinárodné súťaže, preto na sústredenie na základe výsledkov v tejto kategórii nepozývame. Avšak víťaza tejto kategórie čaká hodnotná vecná cena (zvyčajne je to knižka podľa vlastného výberu).

Navyše táto kategória má niekoľko zmien v pravidlách. Riešenia môžete odosielať iba elektronicky a tie sa vám do niekoľkých dní vrátia opravené a obodované. **Pokiaľ ste ale nezískali plný počet bodov, tak môžete skúsiť na riešení popracovať a úlohu poslať znovu** (toto môžete opakovať koľko krát chcete). Navyše finálny termín na odovzdanie je mierne posunutý. Pre príklady v 1. a 2. sérii je to termín odovzdania 2. série a pre príklady 3. a 4. série je to termín 4. série.



Príklady 1. kola zimnej časti

Milé naše riešiteľky, milí naši riešitelia,

Termín odoslania riešení tejto série je pondelok **25. novembra 2010.**

Nevadí, že niečo robím pomaly, lebo možno to robím zle.

KSPáci

1. Zbláznil sa fyzikár...

kat. Z; 10 bodov

Váš učiteľ fyziky sa rozhodol vyskúšať nové pedagogické metódy. Už sa na hodinách skoro nič neučíte, len stále experimentujete a niečo meriate. To mu nemožno až tak zazlievať, aspoň vidíte fyzikálne deje v praxi. No napriek tomu všetci spolužiaci frflú, že sa azda pomiatol, veď zadáva jednu domácu úlohu za druhou.

Posledný mesiac ste strávili neustálym spisovaním protokolov z laboratórnych cvičení. Bolo toho tak veľa, že sa to jednoducho nedalo stíhať – mnoho z nich ste dopisovali narýchlo, tesne pred hodinou. A tak nečudo, že fyzikár nebol s kvalitou spokojný. Údaje nepresné, výpočty chýbajú, tabuľky len krivo rukou načmárané, namiesto aby boli poriadne narysované.

Verdikt: prepracovať! No ešte to tak chýbalo! Čo teraz? Nechce sa vám predsa všetko robiť odznova. Fajn, obsah môžete odpísať od svedomitejších spolužiakov, ale tie tabuľky... tie bude treba narysovať, jednu za druhou. Desiatky tabuliek... nezáživná, zdĺhavá, mechanická práca... priam práca pre robota, nie človeka. Teda, robota alebo... počítač! Pri tejto spásosnej myšlienke sa vám uľavilo. Ešteže viete programovať!

Úloha: Napíšte program, ktorý na vstupe dostane dve prirodzené čísla R a S a vykreslí tabuľku s R riadkami a S stĺpcami. Na vodorovné línie použijete znak '-' (mínus, resp. pomlčka), na zvislé línie znak '|' (tzv. rúra) a na ich priesečníky znak '+' (plus). Každé políčko tabuľky je tvorené jedinou medzerou tesne ohraničenou vertikálnymi a horizontálnymi líniami.

Príklad:

Vstup:

2 3

Výstup:

```
+---++  
| | |  
+---++  
| | |  
+---++
```

2. Zrkadielko, zrkadielko, povedz že mi...

kat. Z; 10 bodov

A je to tu zas. Snehulienkina macocha sa opäť raz pozerá do zrkadla. To síce touto skvostnou panorámou nie je nadšené, nuž ale čo má robiť. Ako má odpovedať? Odpovie, že nie je najkrajšia a... budú z neho tisíce zrkadiel. V rámci záchrany svojho života sa zrkadlo pokúsilo o lacný trik:

„Zrkadielko, zrkadielko, povedz že mi, kto je najkrajší v tejto zemi?“

„Ty, kráľovná. Ale bola by si ešte krajšia, ak by si mala unikátny náhrdelník.“

Kráľovná sa teda rozhodla, že si zabezpečí unikátny náhrdelník. To však nie je také jednoduché. Swarovski majú len malý výber a žien v kráľovstve je veľa. Tadiaľ teda cesta nevedie. A žiadať kráľa o nový náhrdelník je asi ako žiadať žabu o princa.

Po dlhom hneve, počas ktorého zrkadlo skoro stratilo svoju tvár (resp. presnejšie povedané, skoro nové tváre získalo), zrkadlo uznalo, že to možno trochu prehnať. Preto svoj výrok upravilo tak, že stačí, aby bol náhrdelník inak pootočený.

Kráľovná si je teraz istá, že bude najkrajšia na zemi. Potrebovala by však overiť, či je dané otočenie náhrdelníka naozaj unikátne. A na to ho musí najprv vidieť.

Úloha: Na prvom riadku vstupu je reťazec znakov (anglickej abecedy), popisujúci náhrdelník. Na druhom riadku vstupu je jedno celé číslo hovoriace, o koľko treba náhrdelník pootočiť (smerom „doľava“). Náhrdelník sa môže iba otáčať, nemôže sa preklápať (inak by bolo vidno na niektorých šperkoch „made in China“ a to, samozrejme, nechceme).

Na výstup vypíšte jeden riadok – pootočený náhrdelník.

Obmedzenia: Pri tejto úlohe kladieme ešte nasledujúce obmedzenia: váš program si má daný náhrdelník pamätať v jednej premennej typu reťazec, ktorej obsah môže meniť. Nesmiete však používať ďalšie pomocné reťazce ani polia¹. Na výstup vypíšte obsah premennej typu reťazec, do ktorej ste na začiatku načítali náhrdelník zo vstupu (vypíšte ho jediným príkazom, nie po znakoch).

Ak tieto obmedzenia nedodržíte, hrozí vám strata väčšieho počtu bodov...

Kostra riešenia v Pascale:

```
var nahrdelnik: string;
```

```
    K, i, j: integer; {jednoduché pomocné premenné}
```

```
    c: char;
```

```
begin
```

```
    ReadLn(nahrdelnik); {načítame vstup}
```

```
    ReadLn(K);
```

```
    {... otočíme nahrdelnik o K ...}
```

```
    WriteLn(nahrdelnik); {a vypíšeme výsledok}
```

```
end.
```

Príklad:

Vstup:

totojenahrdelnik

4

Výstup:

jenahrdelniktoto

3. Zase ďalšia Alica?

kat. Z; 10 bodov

Alica žila v ríši divov. No dobre, nie tak celkom divov, ale divné to tam bolo až-až. Ani Srdcová kráľovná nebola, čo bývala v rozprávkach. Vlastne, hovorilo sa, že je dosť schopná matematicka. Ale stínanie hláv mala rada presne rovnako, ako sa povrávalo v príbehoch pre deti.

Po tom, čo chytila Alicu, sa rozhodla, že si s ňou namiesto kriketu zahrá svoju obľúbenú hru. Vygenerovala dve náhodné čísla a ku každému náhodný počet výkričníkov². Potom tieto čísla dala svojmu súperovi a ten musel rýchlo povedať, ktoré je väčšie. Počet výkričníkov

¹Obyčajných premenných môžete použiť, koľko sa vám zažiada.

²V skutočnosti jej generátor robili Tidli a Fidli, ktorí mali po sebe napísané čísla a následne tancovali breakdance.

vyjadruje koľko krát sme použili faktoriál. Napríklad platí $3!! = 6! = 720$. A kto neuhádol, tomu dala stať hlavu.

Alica sa preto rozhodla, že vás požiada o pomoc, aby ste jej naprogramovali program, ktorý jej odpoveď povie sám.

Úloha: Na vstupe máte zadané dva riadky. V prvom riadku je číslo x a za ním (po medzere) nejaký počet výkričníkov, v druhom riadku je číslo y a za ním (po medzere) nejaký počet výkričníkov. Máte určiť, ktorý z riadkov zodpovedá väčšiemu číslu a vypísať ho. Ak sú obe čísla rovnaké, vypíšte „rovname“. Môžete predpokladať, že $1 \leq x, y \leq 1\,000\,000$.

Príklad:

Vstup:

3 !!

7 !

Výstup:

7 !

..... Tu sa začínajú zadania KSP-O

4. Zabité dekódovanie

kat. Z a O; 15 bodov

Vesmírna loď Enterprise po dlhej bitke s Borgskou kockou utrpela ťažké poškodenie. Okrem toho, že jej vypadol warpový pohon, tak prestali fungovať aj dekodéry galaktických prenosov. A to je vcelku problém, keďže nemôžu prijímať rozkazy od armádneho velenia³. Preto treba nakódiť nový dekóder.

Galaktické prenosy sú kódované nasledovne. Prenášaný text sa rozseká na slová. Každé slovo má jednoznačne priradený kód (postupnosť 0 a 1). Navyše tieto kódy sú priradené tak, aby žiadny z nich nebol prefixom⁴ iného⁵. A potom sa kódy v jednej súvislej postupnosti prenesú.

Úloha: Na vstupe je číslo M – počet kódových slov. Nasleduje M riadkov tvaru $a\ b$, kde a je nejaké slovo pozostávajúce z písmen malej anglickej abecedy a b je jeho zakódovaný tvar – reťazec z 0 a 1. Môžete predpokladať, že žiadny zakódovaný tvar nie je prefixom iného zakódovaného tvaru. Potom nasleduje číslo N – dĺžka zakódovanej postupnosti a za tým samotná postupnosť z 0 a 1. Vašou úlohou je túto postupnosť dekódovať, teda vypísať na výstup pôvodné slová oddelené medzerou. Môžete predpokladať, že postupnosť sa vždy dá dekódovať.

Môžete predpokladať, že $1 \leq M \leq 1\,000\,000$, $1 \leq N \leq 5\,000\,000$ a že dĺžky slov sú menšie ako 20, dĺžky kódov sú menšie ako 1 000 000 a celkový súčet dĺžok kódov je menší ako 5 000 000.

Odovzdávanie: Tento príklad je praktický, čo znamená, že sa dá odovzdáva jedine elektronicky, pričom stačí odovzdať iba zdrojový kód. Pri odovzdaní sa vám program otestuje na niekoľkých vstupoch a dostane body podľa toho, na koľkých vstupoch dá správny výsledok v časovom limite.

Príklad:

Vstup:

3

slama 111

seno 010

ryba 0111

13

1110100100111

Výstup:

slama seno seno ryba

³A za ich nerešpektovanie kapitán Picard dostane železnou tyčou.

⁴predponou, napr. 011 je prefixom 01100

⁵Toto zaručuje niekoľko prijemných vlastností, napríklad jednoznačné dekódovanie.

5. Otrávený duel

kat. Z a O; 15 bodov

Lord Norton a lord Berkley sa pohádali priam až do krvi a rozhodli sa, že si dajú spravodlivý duel. Lord Norton má však tenisový lakeť v kolene a preto duel nemôže byť v žiadnej fyzickej disciplíne. Preto si vybrali smrteľnú mentálnu disciplínu. Zohnali si špeciálny jed. Pri bežnom jede zomrie ten, kto ten jed vypije. Ale pri tomto jede zomrie ten, čo vypije posledný dúšok.

Ako sa hrá tento smrteľný duel? Zoberie sa jed a rozdelí sa do N pohárov. Týchto N pohárov sa rozdelí do K kruhov. V každom ťahu si jeden z lordov vyberie 1 až L pohárov jedu (nie nutne z toho istého kruhu), vypije ich a kruhy, z ktorých vybral nejaké poháre, spojí do jedného veľkého kruhu. Lordi sa striedajú a nečakane, kto vypije posledný pohár, ten prehrá⁶.

KáeSPáci by radi vedeli, kto vyhrá, ale sú strašné lamy a nevedia si to zistiť. Preto im pomôžte a spravte program, ktorý zistí výsledok.

Úloha: Vašou úlohou je pre dané N , K , L a veľkosti jednotlivých kruhov povedať, ktorý lord vyhrá, ak začína lord Norton a obaja lordi hrajú optimálne.

Príklad:

Vstup:

$N = 21$ $K = 2$ $L = 3$

9 12

Výstup:

Vyhrá lord Berkley

..... Tu sa končia zadania KSP-Z

6. Oliaty koberec

kat. O; 20 bodov

Mali sme raz na T2 modrý koberec. Teraz síce máme stále ten istý koberec, ale už nie je len modrý. Sem-tam sa stane, že ho niekto niečím oleje. A keďže náš koberec nie je najmladší, zažil už niekoľko generácií. A každá z týchto generácií na ňom zanechala nejaké stopy v podobe rôznych flakov.

Halucinka má však rada poriadok, a tak sa rozhodla, že ho trochu vyčistí. Akurát nemala čím. Našťastie v matfyzáckom bufete majú aj čistiace prostriedky. A nie hocijaké. Jeden čistiaci prostriedok vyčistí ľubovoľne veľký flak. A aby toho nebolo málo, v bufete majú aj špiniace prostriedky. Tie fungujú podobne ako čistiace, akurát opačne – zašpinia ľubovoľne veľkú čistú plochu. Jediným problémom je, že sú veľmi drahé (oba rovnako) a treba nimi šetriť. Pomôžte Halucinke vymyslieť, ako vyčistiť koberec tak, aby minula čo najmenej čistiacich a špiniacich prostriedkov.

Úloha: Prvý riadok vstupu obsahuje dve čísla – R S – rozmery koberca. Za tým nasleduje R riadkov dlhých S , ktoré popisujú koberec. Bodka (‘.’) znamená čisté miesto a hviezdička (‘*’) znamená špinavé miesto. Okraj koberca bude zaručene čistý.

Dve susedné špinavé políčka patria do toho istého flaku práve vtedy, keď susedia stranou (iba rohom nestačí). Dve susedné čisté políčka patria do tej istej čistej plochy práve vtedy, keď susedia stranou alebo rohom.

Flak môže obklopovať čistú plochu, ktorá môže obklopovať ďalšie flaky, atď. Žiadny flak ale nebude (priamo) obklopovať viac ako jednu súvislú čistú plochu. Z nasledujúcich situácií je teda prvá povolená, ale druhá nie:

```
*****          *****
* . . . *        * . * . *
* . * . *        * . * . *
* . . . *        * . * . *
*****          *****
```

⁶a zomrel

Napište program, ktorý zistí, koľko najmenej čistiach a špiníach prostriedkov dokopy treba na vyčistenie koberca.

Príklad:

Vstup:

```
5 11
.....
....**....
...*...*.
.***...***
.....
```

Výstup:

```
2
Na koberci sú tri flaky. Zašpiníme ľavý horný
roh koberca, takže bude celý špinavý okrem
jediného štvorčeka. Potom ho naraz vyčistíme.
```

7. Odhaľ komunikačnú sieť!

kat. O; 20 bodov

Tvoja krajina je vo vojne! Má však špióna u nepriateľa. Veliteľstvo by rado zistilo, ako vyzerá komunikačná sieť nepriateľa. Už vie počet a zoznam komunikačných zložiek nepriateľa. Vie dokonca aj to, že komunikácia medzi jednotlivými zložkami prebieha po grafe typu strom (tzn. neobsahuje cykly). Problémom je, že dohodnutá komunikácia s vaším špiónom je obmedzená nasledovne: špióna sa môžete spýtať len to, či sa A a B dohovárajú prostredníctvom C a on vám odpovie áno, ak správy medzi A a B prechádzajú v nepriateľovej sieti cez C .

Úloha: Na vstupe je počet komunikačných zložiek nepriateľa N . Vašou úlohou je spraviť algoritmus, ktorý bude postupne klásť otázky typu (a, b, c) , kde a, b, c sú čísla zložiek nepriateľa ($a, b, c \leq N$). Na základe odpovedí vypíšte zoznam dvojíc, ktoré komunikujú priamo (tzn. nie cez prostredníka). Odpoveď áno dostanete vtedy, ak a, b komunikujú cez c , inak dostanete odpoveď nie (uvedomte si, že stále môžu komunikovať aj cez inú zložku d).

Príklad:

Vstup:

```
 $N = 3$ 
nie      <--
ano      <--
```

Výstup:

```
1 2 3
1 3 2

1 2
2 3
```

8. O dračej invázii

kat. O; 25 bodov

Kde bolo, tam bolo, bolo raz jedno kráľovstvo. V tom kráľovstve bol hrad a mestá pospájané obojsmernými cestami tak, že sa dalo z každého mesta dostať do hradu.

Jedného dňa však priletelo zopár drakov. Každý si vybral nejaké mesto (nie hrad), ktoré spustošil, zožral jeho obyvateľov, usadil sa v ňom a začal zle vplývať na cestovný ruch – cez mesto obsadené drakom totiž nikto živý neprejde.

V niektorých mestách sídlili zemepáni. A práve od nich sa kráľ dozvedel o invázii drakov. Presnejšie, ak sa živý zemepán (to znamená, že ho nezozral drak!) nemohol zo svojho mesta dostať do hradu, poslal kráľovi holubou poštou list. V ňom sa sťažoval na drakov, farbisto opísal svoju situáciu⁷ a žiadal okamžitú nápravu. Žiaľ, zabudol spomenúť, v ktorýchže to mestách vlastne videl drakov.

Pomôžte zhrozenému kráľovi zistiť, aké škody sú spôsobené v tom najlepšom možnom prípade. Najmenej z koľkých miest sa nedá dostať do hradu (vrátane tých obsadených drakmi)?

⁷Óch, ja nešťastný zemepán...

Úloha: Na vstupe dostanete v prvom riadku čísla N (počet miest, $1 \leq N \leq 100\,000$), M (počet ciest, $0 \leq M \leq 1\,000\,000$) a Z (počet správ od zemepánov, $0 \leq Z < N$) oddelené medzerami.

Mestá kráľovstva sú očíslované od 2 do N , hrad má číslo 1. Nasleduje M riadkov v tvare $a\ b$, ktoré popisujú cestnú sieť – obojsmerná cesta spája miesta a a b ($1 \leq a, b \leq N$).

Na poslednom riadku je Z čísel oddelených medzerami – mestá, z ktorých sa ozvali zemepáni. Tieto mestá neboli obsadené drakmi, len sa z nich nedá dostať po cestách do hradu bez navštívenia mesta s drakom.

Vypíšte najmenšie také číslo K , že existuje rozmiestnenie drakov, ktoré spĺňa podmienky zo vstupu (teda že nešťastní zemepáni sa nemôžu dostať do hradu) a počet miest, z ktorých sa nedá dostať do hradu, je práve K .

Odobzdávanie: Tento príklad je praktický, čo znamená, že sa dá odovzdáva jedine elektronicky, pričom stačí odovzdať iba zdrojový kód. Pri odovzdaní sa vám program otestuje na niekoľkých vstupoch a dostane body podľa toho, na koľkých vstupoch dá správny výsledok v časovom limite.

Príklad:

Vstup:

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

Výstup:

```
3
Mesto 2 je určite obsadené drakom; z miest 4
a 5 sa určite nedá dostať do hradu.
```

..... Tu sa končia zadania KSP-O a začínajú zadania KSP-T

9. Tvoriví mládenci

kat. T; 25 bodov

Starí mládenci Bob a Nishrep nie sú takí priemerní ako väčšina mužov v ich veku. Zatiaľ čo ostatní majú štandardné záujmy ako manželky, futbal alebo autá, Boba a Nishrepa spája veľká spoločná vášeň – pestovanie rastlín. Každý z mláďencov má pred svojím domom dlhý chodník, vedľa ktorého sa tiahne krásny rad pestrofarebných kvetov. Táto harmónia farieb a vôní je výsledkom dlhoročnej práce a zodpovedného prístupu.

Bob má vo svojom rade 26 typov kvetov, ktoré si ako správny vyštudovaný programátor označuje malými písmenami anglickej abecedy. Preto si jeho rad môžeme reprezentovať ako reťazec písmen. Zatiaľ čo ľubovoľná nematfyzická dôchodkyňa by svoje štyri kvety v rade vymenovala slovné (levanduľa, astra, margarétka a znova astra), Bob svoju záhradku označuje skrátené **lama**.

Jedného dňa sa Bob rozhodol, že skúsi Nishrepovi povedať presný popis jeho radu kvetov. Ten je ale natoľko dlhý, že starému pestovateľovi robí ťažkosti zapamätanie si už aj jeho písmenkového označenia. Preto sa rozhodol, že si zapamätá len nejakú skomprimovanú verziu, ktorá mu bude stačiť na to, aby u Nishrepa zrekonštruoval celý svoj pás kvetov. Vymyslel si nasledovný, ľahko zapamätateľný, postup: ak **X** je ľubovoľný neprázdny reťazec znakov, tak ak sa v slove za sebou vyskytuje k -krát reťazec **X**, potom si namiesto **XX...X** stačí zapamätáť **k[X]**. Samozrejme, toto môžeme robiť aj rekurzívne. Napríklad reťazec **aaabaaabaaab** môžeme zakódovať ako **3[a]b3[a]b3[a]b**, čím sme dĺžku kódu síce zväčšili, ale môžeme pokračovať ďalej a tento reťazec zakódovať ako **3[3[a]b]** a dostať kód dĺžky 8 (oproti pôvodným 12). Avšak, ak by sme pôvodný reťazec radšej zakódovali ako **3[aaab]**, dostávame dokonca dĺžku 7. Samozrejme, že číslo môže mať aj dlhší dekadický zápis ako jeden znak.

Úloha: Vašou úlohou je pre pôvodný reťazec Bobovej záhrady získať najkratší možný kód v popísanom kódovaní. Inými slovami, vypíšte reťazec pozostávajúci z malých písmen anglickej

abecedy, cifier a hranatých zátvoriek, ktorý sa dekoduje na vstupné slovo a spomedzi všetkých takýchto kódov nájdite najmenší. V prípade viacerých možností vypíšte ľubovoľnú.

Príklad:

Vstup:

abcabcabc

Výstup:

abca3[bc]

Ďalší zaujímavý spôsob zakódovania by bol 2[abc]2[bc], avšak ten je o dva znaky dlhší.

10. Teória výberu suseda

kat. T; 25 bodov

Prišiel raz MišoF v polke veľkého stretka do KSPárne a pozerá na N diskutujúcich ľudí. Žiadny KSPák sa na veľkom stretku okrem mentálnych kvalít nezaobíde bez troch vecí: tácky s hranolkami, pohára kofoly a tabuľky čokolády. Každý účastník stretka má nejakú zásobu každej z týchto troch vecí. Tieto zásoby budeme reprezentovať prirodzenými číslami – v prípade čokolády je to počet kociek, v prípade hranoliek je to počet hranoliek a v prípade kofoly je to počet MišoFových glgov.

MišoF popri bočných úmysloch (napríklad povedať niečo múdre) sleduje svoj primárny cieľ – k jednej osobe si prisadnúť a zjesť jej všetky hranolky, všetku čokoládu a vypíť jej celú kofolu. Začal sa obzerať po ľuďoch a zisťovať, kto má koľko akej potraviny. MišoF je pre účely tejto úlohy parametrizovateľný troma reálnymi nezápornými hodnotami – momentálnou chuťou na každú z potravín. Preto začal rátať takzvanú *príjemnosť posedenia* pre každého človeka. *Príjemnosť posedenia* s človekom je daná ako súčet hodnôt jeho potravín, pričom hodnota potraviny človeka je súčin jej množstva a MišoFovej chuti na danú potravinu. MišoF si vyberie človeka s najvyšším koeficientom *príjemnosti posedenia*.

Keď to ale MišoF všetko porátal, uvedomil si, že si chce z istých príčin sadnúť vedľa Ušámu. Nechce ale zradiť svoju oficiálnu a verejnú stratégiu výberu a preto sa rozhodol, že upraví svoje chute na potraviny. Pomôžte MišoFovi a nájdite hodnoty chutí na kofolu, hranolky a čokoládu tak, aby bol pre neho najvýhodnejší sused Ušáma (prípadne zistíte, že pri daných množstvách to nie je možné). Pomôžte mu rýchlo, pretože ináč Ušáma stihne zjesť svoje hranolky a načo to potom bude dobré?

Úloha: Na vstupe je dané číslo N . Nasleduje N trojíc prirodzených čísel – koľko majú jednotliví ľudia hranoliek, čokolády a kofoly. Prvá trojica sú Ušámove hodnoty, $N - 1$ zvyšných KSPákov nie je dôležité menovať. Čísla, ktoré vypíšete, nemusia byť celé (majú byť ale nezáporné).

Príklad:

Vstup:

$N = 3$

1 2 5

3 3 3

4 1 5

Výstup:

1 4 5

Pri tomto ohodnotení bude mať Ušáma hodnotu príjemnosti posedenia 34, zatiaľ čo ostatní dvaja KSPáci len 30, respektíve 33.