

# Korešpondenčný seminár z programovania

Leták letnej časti XXX. ročníka

**Korešpondenčný seminár z programovania (KSP)** je súťaž programátorov – stredoškolákov a mladších – pripravovaná skupinou študentov a doktorandov FMFI UK. Jej cieľom je zdokonaľiť žiakov v programovaní a v algoritmickej mysli.

Ak študuješ na strednej škole a vieš aspoň trochu programovať, neváhaj a zapoj sa do našej súťaže, má to množstvo výhod:

- Riešením súťažných úloh a štúdiom našich vzorových riešení sa môžeš naučiť mnoho nového. Získané poznatky a skúsenosti sa ti iste budú hodiť v iných súťažiach v programovaní (napríklad pri riešení Olympiády v informatike), počas vysokoškolského štúdia, či pri prijímacích pohovoroch do zamestnania. (Mnoho našich bývalých riešiteľov sa bez ťažkostí zamestnalo v špičkových IT spoločnostiach ako Google, Facebook, ESET, ...)
- 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 považuje za zaujímavé.
- Pre najlepších riešiteľov organizujeme každoročne dve týždenné sústreďenia. Sústreďenie je jedinečnou príležitosťou ako spoznať nových priateľov s podobnými záujmami, naučiť sa čosi viac nielen o programovaní a zažiť kopeček zábavy.

## Ako KSP prebieha?

Počas školského roka prebehnú dve samostatné časti súťaže: zimná a letná. Každá časť sa skladá z dvoch kôl, každé kolo obsahuje desať súťažných úloh. Najlepších riešiteľov zimnej časti pozývame na jarné sústreďenie; najlepších riešiteľov letnej časti zase na to jesenné.

Súťažiť sa dá v troch kategóriách: **Z** (pre začínajúcich riešiteľov, obsahuje úlohy 1–5), **O** (skúsenejší riešitelia, úlohy 4–8) a **T** (špeciálna kategória pre náročných, päť samostatných úloh). Každá kategória má svoju vlastnú výsledkovú listinu. Na sústreďenia pozývame riešiteľov na základe výsledkov v kategóriách Z a O.

Vaše riešenia úloh môžete odovzdávať na stránke <http://www.ksp.sk/eRiesenie>, kým neuplynú termín určený v zadaniach kola. Po tom, čo riešenia opravíme, nájdete na tomto mieste aj naše komentáre k nim a počet bodov získaný za jednotlivé úlohy.

## Ako má vyzeráť riešenie a za čo dostanem body?

Vašou úlohou je vytvoriť program, ktorý rieši zadanú úlohu. V prvom rade sa snažte, aby bol korektný, t.j. aby dal pre každý vstup správnu odpoveď, v druhom rade aby bol čo najrýchlejší a mal čo najmenšie pamäťové nároky.

Riešenie úlohy pozostáva z programu a popisu použitého algoritmu. V zadaní vždy uvedieme, koľko bodov sa dá získať za program a koľko za popis; výsledný počet bodov za úlohu je súčtom týchto dvoch hodnotení.

Váš program hneď po odovzdaní automaticky otestujeme na viacerých vopred pripravených vstupoch. Body za neho vám pridáme podľa toho, na koľkých vstupoch dá správnu odpoveď v časovom limite. Len čo sa program dotestuje, dozviete sa výsledok. Ak ste nezískali plný počet bodov, môžete program vylepšiť a odovzdať ho znova. Podrobnejšie informácie o odovzdávaní programov nájdete na našej webstránke.

Popis algoritmu by mal byť natoľko podrobný a zrozumiteľný, aby bolo možné podľa neho napísať program rovnako efektívny, ako ten váš. Ďalej vyžadujeme odhad časovej a pamäťovej zložitosti a zdôvodnenie (ak je to potrebné, aj dôkaz) správnosti algoritmu.

Ak vo svojom riešení používate zložitejšie dátové štruktúry (napríklad haldu, nie obyčajné pole), musíte popísať aj ich implementáciu. To platí aj v prípade, že ich váš programovací jazyk už obsahuje a vy ste ich neimplementovali. Ak si nie ste istí, či niečo môžete použiť bez popisu, radšej to popíšte, prípadne sa spýtajte vo fóre na stránke.

Popis odovzdávajte vo formáte PDF alebo ako plain text. Hodnotíme hlavne korektnosť algoritmu a v druhom rade jeho efektívnosť. Získaný počet bodov sa dozviete, keď vaše riešenie po termíne odovzdania opravíme.

Zopár ukázkovo vyriešených starších úloh nájdete na stránke <http://www.ksp.sk/wiki/Seminar/Riesenie>. Na tomto mieste sa tiež môžete dočítať, čo je vlastne časová a pamäťová zložitosť (ak vám tieto pojmy veľa nehovoria).

### Ktoré kategórie môžem riešiť?

Kategóriu Z môžu riešiť:

- tretiaci a štvrtáci<sup>1</sup>, ak do začiatku príslušného polroka neboli na sústreďení KSP,
- druháci a mladší, ak do začiatku príslušného polroka boli najviac na jednom sústreďení KSP.

Kategórií O a T sa môžu zúčastniť všetci stredoškóľáci (a mladší) bez obmedzenia. Ak vám to pravidlá dovoľujú, môžete riešiť aj viacero kategórií naraz.

### Čo je na kategórii T iné?

Keď sa vám zdá, že príkladov je málo, chcete sa naučiť viac, osvojiť si nové finty a vyskúšať si naprogramovať niečo, čo ste doteraz neskúšali, táto kategória je akurát pre vás.

Úlohou tejto kategórie je aj celoročná príprava riešiteľov na medzinárodné súťaže. Jej víťaza čaká večná sláva a navyše **bodý z kategórie T budú mierne zohľadnené pri výbere reprezentácie na Medzinárodnú olympiádu v informatike**. (Tento výber má formu týždňového sústredenia, na ktoré sú pozvaní najlepší riešitelia celoštátneho kola Olympiády v informatike, kat. A.)

Na rozdiel od kategórií Z a O, **v kategórii T nemusíte písať popis riešenia, odovzdávate len program. Zadania kategórie T už nenájdete v papierovej podobe, ale len na internetovej stránke <http://www.ksp.sk/wiki/Zadania/Zadania>**.

Na našej stránke sa dozviete aj viac o tejto kategórii.

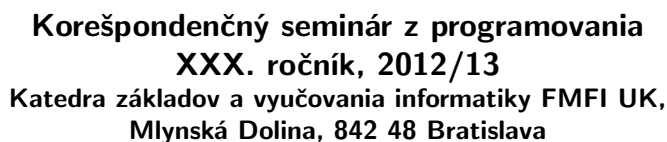
### Registrácia

Pred odovzdaním elektronického riešenia je potrebné zaregistrovať sa na našej webstránke 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).

Účasťou v KSP nám dávate súhlas spracovať a archivovať údaje, ktoré nám poskytnete pri registrácii, ako aj zverejniť vaše meno, školu, ročník a získané body vo výsledkovej listine.

---

<sup>1</sup>Za štvrtákov považujeme študentov, ktorí maturujú v tomto školskom roku; tretiaci sú tí, ktorí budú maturovať budúci školský rok; ostatné ročníky analogicky.



**Termín** odoslania riešení tejto série je pondelok **25. marca 2013.**

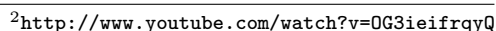
kat. Z; 7 b za popis, 3 b za program

Teraz má Maru doma hrbu fotiek zázračného autobusu. Rozhodla sa, že si urobí štatistiku, koľkokrát ho od fotila, keď vylietaval a koľkokrát, keď vlietaval do garáže. Tých fotiek je však strašne veľa a Maru ako vždy nemá čas. Preto vás žiada o pomoc. Pomôžte Maru rozhodnúť, či autobus na fotke vlietaval, alebo vylietaval z garáže. Bude stačiť ak zistíte, či letel doprava alebo doľava.

V prvom riadku vstupu budú dve celé čísla  $r$  a  $s$  ( $1 \leq r, s \leq 1000$ ) udávajúce počet riadkov a písmen v riadkoch, ktoré tvoria obrázok autobusu. Ďalej nasleduje  $r$  riadkov. V každom riadku je presne  $s$  znakov. Znak, ktorý môže byť v obrázku, znamená toto:  $/$ ,  $|$ ,  $\backslash$ ,  $-$  sú hrany karosérie,  $.$  je plocha karosérie alebo pozadie,  $0$  je koleso,  $o$  je okno a  $H$  sú dvere. Žiaden iný znak sa tam nevyskytne.

Vypíšte slovo **doprava**, ak autobus letí doprava, prípadne slovo **dolava**, ak letí dolava.

doprava



vstup

```

7 25
.....|.....
....._|...|.....
...../.oo.oo.oo.HH.oo\..
.|.\_/.oo.oo.oo.HH.oo\..
.|.|_.....HH....|.
.|_/.\_____/.
.....

```

výstup

doprava

*Zázračný školský autobus sa vie premeniť napríklad aj na ponorku.*

## 2. Známká

kat. Z; 7 b za popis, 3 b za program

Žaba na Vianoce dostal stroj času, a tak sa ho ešte cez prázdniny rozhodol vyskúšať. Keďže má vyššie ciele ako napríklad zabiť Hitlera, premiestnil sa do času pred štyrmi rokmi, keď ešte usilovne študoval na gymnáziu v slávnom Lučenci.

Tam sa úspešne vlámал do budovy školy s jedinou myšlienkou v hlave: nájsť klasifikačný hárok a prepísať niekoľko známok v ňom tak, aby mu na polrok vychádzala z nemčiny „čistá dvojka“ – teda aby jeho priemer bol presne 2.0. Tak totiž docieli, že vtedajší Žaba dostane bez ďalšieho skúšania dvojku. (No a to, aby sa vtedajší Žaba vyhol skúšaniam a pritom dostal známku, ktorá je dosť dobrá, no nie až podozrivo dobrá, je veľmi podstatné pre správny chod vesmíru.)

### Úloha

Na vstupe dostanete počet jednotiek, dvojok, trojok, štvoriek a päťiek v klasifikačnom hárku. Zistíte, koľko najmenej z nich treba zmeniť (t.j. prepísať na ľubovoľnú známku), aby ich priemer bol presne 2.

### Formát vstupu

V jedinom riadku vstupu je päť čísel:  $z_1, \dots, z_5$ , pričom  $z_i$  je počet známok s hodnotou  $i$ , ktoré má Žaba v klasifikačnom hárku. Súčet počtov známok neprekročí  $10^{10}$ .

**Pozor!** Klasické premenné, `longint` v Pasmale a `int` v C++, majú obmedzenú veľkosť, čiže sa do nich zmestia čísla len od  $-2^{31}$  do  $2^{31} - 1$  ( $-2\,147\,483\,648, \dots, 2\,147\,483\,647$ ) vrátane. Čísla na vstupe môžu presiahnuť túto veľkosť. Použite `int64` resp. `long long`.

### Formát výstupu

Vypíšte jeden riadok a v ňom jedno celé číslo: najmenší počet známok, po zmene ktorých bude Žabov priemer presne 2.0.

### Príklady

vstup

```
1 2 1 0 2
```

výstup

```
2
```

*Priemer je pred zmenou 3.0. Na priemer 2.0 stačí zmeniť napríklad obe päťky na dvojky.*

vstup

```
3 1 0 0 0
```

výstup

```
1
```

*Priemer je pred zmenou 1.25. Priemer 2.0 sa dá dosiahnuť napríklad zmenou jednej jednotky na štvorku.*

## 3. Zemčo a trojuholníky

kat. Z; 8 b za popis, 4 b za program

Malý Zemčo má doma veľkú krabicu s paličkami a veľmi rád si z týchto paličiek skladá trojuholníky.

Napríklad vyberie z krabice paličky s dĺžkami 3, 4 a 5 a postaví trojuholník – čiže, tento je dokonca pravouhlý! Alebo môže vybrať paličky s dĺžkami 1, 4 a 9... ale fuj, z nich sa trojuholník poskladať nedá. Keby sa o to pokúšal, mohol by si vypichnúť oko.

My samozrejme nechceme, aby si Zemčo ublížil. Preto mu chceme z krabice nejaké paličky povyhadzovať, aby v krabici neostala žiadna trojica, z ktorej by sa nedal postaviť trojuholník.

A bolo by fajn, aby mu v krabici ostalo čo najviac paličiek, nech sa má s čím hrať.

## Úloha

Na vstupe dostanete zoznam dĺžok paličiek v Zemčovej krabici. Keďže má Zemčo vo svojej krabici poriadok, dĺžky na vstupe sú zadane v neklesajúcom poradí.

Nájdite a vypíšte najmenší možný počet paličiek, ktoré treba z krabice vyhodíť, aby si určite nevypichol oko – teda aby sa z každej trojice paličiek, ktoré ostanú, dal poskladať trojuholník.

## Formát vstupu

Na prvom riadku vstupu je číslo  $n$  ( $1 \leq n \leq 200\,000$ ), počet paličiek v krabici. Na druhom riadku je vymenovaných  $n$  kladných celých čísel  $x_1 \leq x_2 \leq x_3 \leq \dots \leq x_n$ , pričom  $x_i$  je dĺžka  $i$ -tej paličky a  $1 \leq x_i \leq 10^9$  pre všetky  $i$ .

## Formát výstupu

Na výstup vypíšte jedno číslo – koľko najmenej paličiek musíme z krabice odstrániť.

## Príklady

vstup

```
4
2 2 4 4
```

výstup

```
1
```

*Trojica (2, 2, 4) je nebezpečná, ale ak vyhodíme jednu dvojku, tak bude krabica v poriadku.*

vstup

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

výstup

```
2
```

*Napríklad vyhodíme 1 a 2.*

vstup

```
3
1 4 9
```

výstup

```
1
```

*Z dvoch paličiek sa nedá poskladať žiaden trojuholník, takže stačí vyhodíť ľubovoľnú paličku.*

## 4. Zahrajme sa

kat. Z a O; 10 b za popis, 5 b za program

Baška s Olíviou sú najlepšie kamarátky. Snažia sa maximalizovať čas, ktorý spolu trávia, minimalizovať rozdiely v odevoch, ktoré na sebe nosia, a čo najviac zosynchronizovať svoju chôdzu. Toto všetko je únavné, a preto musia aj oddychovať. Samozrejme, spolu. Medzi najlepšie spôsoby oddychu patria seriály a online hry.

Nedávno prepadli síce otrepanej, ale stále zaujímavej<sup>3</sup> Block N Roll. V tejto hre ovládate kváder, ktorý musíte „preváľať“ na určité miesto, pričom si treba dávať pozor na diery. Keď ich po čase začalo hranie hry nudiť, rozhodli sa, že si spravia program, čo bude hru riešiť namiesto nich. Tak si obe sadli, obe nakódili a čo čert nechcel, ich programy vypisujú rozdielne výsledky. Teraz sa nevedia rozhodnúť, ktorá spravila chybu. Pomôžte im!

## Úloha

Máme obdĺžnikovú mriežku s  $r$  riadkami a  $s$  stĺpcami. Niektoré políčka mriežky chýbajú – sú to diery. Rovnako sa za diery považujú všetky políčka mimo mriežky. Práve jedno políčko mriežky, ktoré nie je dierou, obsahuje červené tlačítko. V ľavom hornom rohu je na mriežke položený kváder s rozmermi  $a \times b \times c$ . Kváder môžete presúvať váľaním, teda tak, že si vyberiete jednu z jeho 4 spodných hrán a prevalíte ho okolo tejto hrany na bočnú stenu. Môžete to však urobiť iba vtedy, ak sa pod stenou, na ktorú kváder prevalíte, nebude nachádzať žiadna diera. Takéto prevalenie sa nazýva ťah. Hru vyhráte, ak spravíte niekoľko (možno aj nula) ťahov tak, aby ste kvádom zavalili políčko s červeným tlačítkom. Vašou úlohou je zistiť, na koľko najmenej ťahov sa daná hra dá vyhrať.

Čiastočné body (za každú podúlohu jeden) bude môcť váš program získať aj za vyriešenie dvoch podúloh. V prvej podúlohe bude platiť  $a = b = c = 1$ . V druhej podúlohe môžete predpokladať, že hra sa dá vyhrať na najviac 10 ťahov.

<sup>3</sup>pre účely programovania

## Formát vstupu

Prvý riadok vstupu obsahuje dve medzerou oddelené kladné celé čísla  $r$  a  $s$  ( $rs < 100\,000$ ) – rozmery mriežky. Druhý riadok vstupu obsahuje tri medzerou oddelené kladné celé čísla  $a$ ,  $b$  a  $c$  ( $a, b, c < 10^9$ ) – rozmery kvádra. Každý z nasledujúcich  $r$  riadkov obsahuje reťazec dlhý  $s$  znakov – popis mriežky. Znak „#“ symbolizuje dieru, znak „X“ symbolizuje červené tlačítko a znak „.“ symbolizuje obyčajné políčko. Na začiatku kváder leží na horných  $b$  riadkoch a ľavých  $a$  stĺpcoch mriežky, pričom je vysoký  $c$  políčok. Môžete predpokladať, že na začiatku kváder neleží na diere.

## Formát výstupu

Na výstup vypíšete jediné číslo – minimálny počet ťahov potrebný na výhru; alebo „-1“, ak sa hra nedá vyhrať.

## Príklad

vstup

```
7 5
1 1 2
.....
...#.
.##..
.##..
.....
##..#
##..X
```

výstup

14

*Kváder preválíme: vpravo-dole-vľavo-dole-dole-  
vpravo-vpravo-hore-vpravo-dole-vľavo-dole-dole-  
vpravo.*

vstup

```
5 5
1 1 2
...#.
.....
.#.#.
..#..
...X
```

výstup

-1

*Nech sa snažíme akokoľvek, nevieme kvádom zavalíť  
„X“ bez toho, aby sme pri tom nevypadli z mriežky.*

## 5. Odchod o 5 minút

kat. Z a O; 0 b za popis, 12 b za program

Skôr či neskôr sa každý z nás dostane do situácie, kedy by sa nám zišiel program, ktorý ako vstup dostane dáta odniekiaľ z webu. Lenže na webe často tie dáta nemáme v žiadnej peknej podobe: skrývajú sa pred nami rozsypané po webstránke. Ako ich odtiaľ pozbierať?

Občas sa medzi nami nájde idealista, ktorý vyhlási, že by sme mohli skúsiť nejaký XML parser. Lenže 99% webu štandardy nevidelo ani z rýchlika a do validného HTML majú ďaleko. Opační extrémisti zase toho čo najviac ručne nadráťujú do programov ako rôzne konštanty. A potom stačí jedna zmena na stránke a môže začať odznova.

V praxi je podľa nás dobrým kompromisom použitie regulárnych výrazov. Pomocou nich si vieme pomerne bezbolestne nájsť na stránke tie dáta, ktoré nás zaujímajú. A presne to si v tejto úlohe vyskúšate aj vy. Alebo aj nie, je len na vás, akou cestou sa vydáte.

Dáta, ktoré nás zaujímajú, sú popisy konkrétnych ciest mestskou hromadnou dopravou. Už sme si ich stiahli na svoj počítač, máme tam kopu súborov, a tie teraz treba spracovať do podoby prítulnejšej pre programy.<sup>4</sup>

Každý súbor, ktorý máme, obsahuje niekoľko popisov cesty. Popis cesty sa skladá z jedného alebo viacerých presunov rôznymi linkami MHD. Každý takýto presun chceme zapísať ako tri riadky textu: v prvom riadku číslo linky, v druhom čas (v 24-hodinovom formáte HH:MM, ako na stránke) a zastávka odkiaľ, v treťom čas a zastávka kam daným spojom ideme.

<sup>4</sup>Sami si podobné súbory môžete stiahnuť zadáním vhodnej adresy s názvami zastávok. Zo shellu to ide napr. príkazom:  
`wget http://imhd.zoznam.sk/ba/planovac-cesty-vyhľadanie-spojenia.html?spojenieodkial=lubovn&spojeniekam=botan`  
ale nezabúďte zbytočne ich server, na riešenie tejto úlohy to vôbec nie je potrebné.

Výstupom spracovania súboru by teda mal byť textový súbor, v ktorom budú postupne za sebou všetky popisy ciest, ktoré sa nachádzajú vo vstupnom súbore. Popis cesty tvorenej  $k$  presunmi má presne  $3k$  riadkov. Medzi jednotlivými popismi ciest ponechajte voľný riadok.

### Testovacie dáta a písanie riešení

Aby ste si svoje programy mohli odladiť, pripravili sme pre vás sadu vstupov a zodpovedajúcich výstupov. Môžete si ich stiahnuť z <http://www.ksp.sk/wiki/uploads/Zadania/ksp30-3-5-sample.zip> a svoj program na nich otestovať.

### Odovzdávanie riešení

Odovzdávate normálne program ako v iných úlohách KSP. Na štandardný vstup dostane text stránky, na štandardný výstup má vypísať správne formátované popisy ciest ktoré obsahovala stránka zo vstupu.

Stránky použité pri hodnotení vašich riešení budú iné od tých, ktoré sme vám dali, ale budú pochádzať z toho istého zdroja. Budú rozdelené do troch rôznych ťažkých sád, za správne riešenie každej sady sú 4 body.

V prvých dvoch sádach sa ako začiatočná, koncová, aj ako prestupné zastávky vyskytujú len zastávky, ktoré nemajú v mene diakritiku. V prvej sade navyše platí, že možností cesty je presne 5 a že všetky sú priame (bez prestupu).

## 6. Olívia vstupuje na scénu

kat. O; 13 b za popis, 7 b za program

Mnohí z vás ešte nepoznajú Olíviu. Je to milá, kučeravá baba, nemá však úspech u opačného pohlavia. Na Olíviu bola totiž uvalená strašlivá kliatba. Každý chalan, čo ju pozná, sa ju snaží zbaliť, ale keď oňho prejaví záujem ona, nájde si inú. Naposledy sa to Olívii stalo s Bobom. Hneď ako po ňom hodila očkom, Bob si našiel Mariku. To už bolo aj na ňu priveľa, preto sa rozhodla odísť z matfyzu a pridať sa do tajného Kultu Skrytého Poznania.

Kult Skrytého Poznania vychováva tých najlepších umelcov, svetobežníkov a všemumelcov na celom svete. Keď prejdete ich tréningom a dostanete Čestný Rád Všumelca, ste schopný nevidaných skutkov. Zamestnať sa v ľubovoľnej práci, vytvoriť kameň mudrcov či sfalšovať van Goghove Slnčnice, proste stať sa legendou. Aj známy Phileas Fogg, ktorý obišiel celú zemeguľu iba za 80 dní, bol váženým členom KSP.

Aby ste sa dopracovali až k vysväteniu, čaká vás veľmi komplikovaná cesta, plná nebezpečného a náročného tréningu a rôznych skúšok. Uchádzači musia totiž fyzicky aj mentálne prevyšovať schopnosti obyčajných smrteľníkov. Musia plynule ovládať niekoľko desiatok jazykov, poraziť v zápase bojovníka suma, ovládať maticový počet, vedieť vyrobiť bombu aj z výkalov... Už snáď máte predstavu o tom, aké náročné je stať sa čestným členom. Dokonca ani MacGyver sa nedostal až k vysväteniu, lebo neprešiel skúškou z matematickej analýzy.

Olívia si teda podala žiadosť do tejto tajnej spoločnosti a jedného januárového rána ju zobudil hluk helikoptéry, ktorá ju prišla vyzdvihnúť. Spustila jej lano, na ktoré Olívia naskočila, a odviezla ju až do Vysokých Tatier. Tam totiž na Olíviu čakala prvá skúška, ktorá rozhodne o tom, či Olívia bude prijatá na zácvik, alebo ju nechajú zahynúť – v Kulte sa s človekom nemaznajú. Ak sa k nim chcete pridať, musíte riskovať.

Skúška vyzerala nasledovne: vysoko na tatranských štítoch boli v rade umiestnené plošinky, kde vždy dve susedné plošinky boli prepojené pomocou lanového mosta. Mosty sú však veľmi chatrné a  $i$ -ty z nich vydrží len  $a_i$  prechodov. Potom sa rozpadne a už sa nebude dať chodiť z jednej strany na druhú. Olíviinou úlohou je vybrať si niektorú plošinku, na ktorú ju helikoptéra zosadí a následne sa presúvať po mostoch medzi plošinami tak, aby sa presunula čo najviackrát – presun znamená, že sa pohne z jednej plošiny na susednú použitím mosta, ktorý sa ešte nerozpadol. Keď sa ocitne na plošine, z ktorej nepovedie už žiadny most, senseji z Kultu sa rozhodnú, či sa nedalo po mostoch presúvať dlhšie, a ak nie, zoberú Olíviu do ich tajného chrámu. Inak ju tam nechajú, čo by pre Olíviu znamenalo nevyhnutný koniec.

To však Olívii nehrozilo, rýchlo našla cestu, ktorou sa presunie najviackrát a za polhodinu už odletela helikoptérou v ústrety vychádzajúcemu slnku. Otázkou však zostáva, ako by ste pochodili na jej mieste vy.

### Úloha

Máte zadané číslo  $n$ , počet plošín. Plošiny sú zoradené v rade a medzi plošinami  $i$  a  $i + 1$  je natiahnutý most, ktorý vydrží  $a_i$  prechodov, kým sa nerozpadne. Vašou úlohou je zistiť najväčší počet prechodov cez mosty, ak môžeme začínať na ľubovoľnej plošine.

### Formát vstupu

Na prvom riadku je číslo  $n$  udávajúce počet plošín, pričom  $2 \leq n \leq 10^5$ . Na druhom riadku je  $n - 1$  kladných čísel – počet prechodov, ktoré vydrží daný most. Čísla sú zadané ako  $a_1, a_2, \dots, a_{n-1}$ , pričom môžete

predpokladať, že každé z týchto čísiel je menšie ako  $10^9$ .

### Formát výstupu

Vypíšte jedno číslo – najväčší počet presunutí medzi plošinami.

### Príklad

vstup	výstup
5 2 1 2 1	5

*Môžeme začať na plošine 3 a postupne sa presunúť na plošiny 4, 3, 2, 1, 2. Po týchto pohyboch je jediný nezničený most medzi plošinami 4 a 5, tam sa však nevieme dostať a preto končíme.*

## 7. Odfotiť si škôlku

kat. O; 13 b za popis, 7 b za program

Na Veselej ulici stojí jedna smutná škôlka. Prečo je smutná? Na deti tam totiž dozerá prísna pani učiteľka so svojráznym zmyslom pre estetiku. Dnes si napríklad zaumienila, že urobí škôlkarom skupinovú fotografiu. Deti by na nej nemali postávať len tak hocijako, ale v jednom peknom, dlhom rade. Pani učiteľka si dokonca vymyslela aj presné poradie, v akom majú deti stáť, aby dosiahla skutočne harmonickú kompozíciu výslednej fotografie.

Ako sa však dalo čakať, neposednú bandu škôlkarov fotenie vôbec nezaujímalo. A tak skôr než pani učiteľka stihla stlačiť spúšť, niektoré deti opustili svoje miesta v rade, trochu sa pomotali, premiešali a potom sa rýchlo opäť zaradili. Nemuseli sa však vrátiť na svoje pôvodné pozície, dokonca sa ani nemuseli vrátiť do vzniknutých medzier – niektoré sa mohli vtlačiť medzi deti, ktoré poslušne zostali v rade stáť.

Pani učiteľka tak získala fotografiu, ktorá vôbec nebola podľa jej predstáv. Nahnevaná, no odhodlaná splniť svoj cieľ, opäť rozostavila deti do harmonického poradia, vyťahala vinníkov za uši a vrátila sa k fotoaparátu. Ani ďalšia fotka sa však nepodarila: Potrestané deti síce zostali poslušne v rade, ale našla sa iná skupina galganov, ktorí sa podobne ako tí pred nimi premiešali a nesprávne zaradili.

Toto celé sa opakovalo dovtedy, kým pani učiteľka nestratila trpezlivosť a vedomie, teda dokopy päťkrát. Rodičia škôlkarov sa ju teraz chystajú navštíviť v nemocnici a ako darček jej plánujú priniesť jej vytúženú fotografiu s deťmi nastúpenými v harmonickom poradí. Ani oni síce netušia, ako udržať malých nezbedníkov stáť bez pohnutia, no zatiaľ riešia pálčivejší problém: Ako to harmonické poradie vlastne vyzerá?

### Úloha

V škôlke je  $n$  detí. Keďže sa podobajú ako vajce vajcu, označme si ich číslami od 1 do  $n$ . Harmonické poradie je nejaká permutácia detí, označme si ju  $\pi$ .

Dostanete zadaných päť permutácií čísel  $1, 2, \dots, n$ . Každá z nich vznikla z  $\pi$  tak, že sme niekoľko (možno nula) prvkov odstránili a potom ich povkladali späť (možno na iné pozície). Navyše pre každé číslo platí, že sme ho odstránili a znovu vložili najviac v jednej z týchto piatich permutácií.<sup>5</sup>

Vašou úlohou je nájsť permutáciu  $\pi$ . Ak existuje viacero vyhovujúcich, vypíšte ľubovoľnú. Môžete predpokladať, že existuje aspoň jedno riešenie.

### Formát vstupu

V prvom riadku vstupu je zadané jedno celé číslo  $n$  ( $1 \leq n \leq 100\,000$ ). Každý z nasledujúcich piatich riadkov obsahuje permutáciu čísel  $1, 2, \dots, n$  oddelených medzerami.

### Formát výstupu

Vypíšte jeden riadok a v ňom permutáciu  $\pi$ . Jednotlivé čísla oddeľujte medzerou.

### Príklad

<sup>5</sup>To ešte neznamená, že sa každé číslo nachádza na zlom mieste najviac v jednej permutácii. Prvok sa môže premiestniť napríklad aj tak, že odstránime nejaké prvky pred ním a vložíme ich za neho.



vstup

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

výstup

```
1 3 5 2 4
Dieťa 1 sa presunulo na tretej fotke, dieťa 4 na piatej
fotke, deti 3 a 5 sa presunuli na druhej fotke.
```

## 8. Otravná blbosť

kat. O; 15 b za popis, 10 b za program

Tommy, znechutený z práce pre Vlejda a Hermiho vo firme Macher,<sup>6</sup> smutne skonštatoval „Blbosť som si vybral!“ a šiel si hľadať inú prácu.

Nečudo, veď jeho posledná úloha bola úplne otravná blbosť, ktorá spočívala v prevádzaní jedného čísla na druhé pomocou veľmi čudných operácií. Čo je smutné, po odchode Tommyho to vo firme Macher nemá kto vykonávať a tak to ostalo na vás.

### Úloha

Máte zadané tri kladné celé čísla  $a, b, k$ , kde  $a \geq b$ . Číslo  $a$  môžete meniť pomocou nasledujúcich operácií:

- Odčítaj 1 od čísla  $a$ .
- Vyber si  $x$  také, že  $2 \leq x \leq k$  a odčítaj od  $a$  číslo  $(a \bmod x)$ .

Vašou úlohou je zmeniť číslo  $a$  na číslo  $b$  s použitím najmenšieho počtu operácií. **Dajte si záležať pri dôkaze správnosti vášho algoritmu.** Správnosť a formálnosť dôkazu bude výrazne ovplyvňovať počet bodov získaných za popis riešenia.

### Formát vstupu

Na vstupe je jeden riadok, ktorý obsahuje tri kladné celé čísla  $a, b, k$  oddelené medzerou. Platí:  $1 \leq b \leq a \leq 10^{18}$  a  $2 \leq k \leq 15$ .

### Formát výstupu

Vypíšte jeden riadok a v ňom jedno číslo: minimálny počet operácií nutný na pretransformovanie čísla  $a$  na číslo  $b$ .

### Príklad

vstup

```
9 1 4
```

výstup

```
6
```

Dá sa to napríklad takto: 9, 8, 6, 4, 3, 2, 1.

vstup

```
7 3 10
```

výstup

```
2
```

Ide to napríklad takto: 7, 4, 3.

vstup

```
1000000000000000000 1 4
```

výstup

```
5000000000000000001
```

# Zadania kategórie T

Zjavia sa na našej stránke <http://www.ksp.sk/wiki/Zadania/Zadania> ešte v januári. Nezabudnite sa na ne pozrieť, čaká na vás ďalších **päť** zaujímavých úloh rôznych obtiažností.

<sup>6</sup>odvodené od Macko a Herman