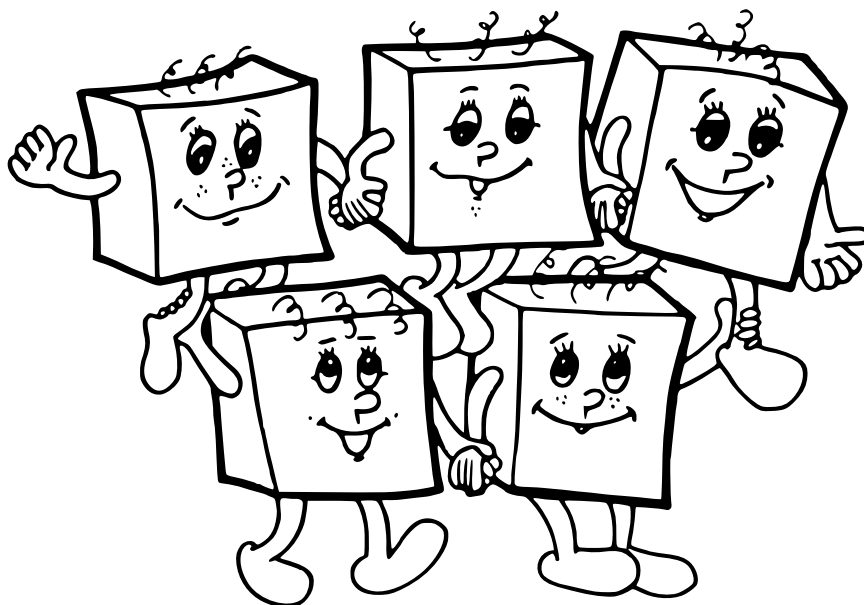


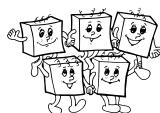
# OLYMPIÁDA V INFORMATIKE NA STREDNÝCH ŠKOLÁCH



**dvadsiaty piaty ročník**  
školský rok 2009/10

**zadania domáceho kola**  
**kategórie A a B**

- **Olympiáda v informatike** je od školského roku 2006/07 samostatnou súťažou. Predchádzajúcich 21 ročníkov tejto súťaže prebiehalo pod názvom **Matematická olympiáda, kategória P** (programovanie).
- Oficiálnu **webstránku** súťaže nájdete na <http://oi.sk/>.



## Informácie a pravidlá

### Pre koho je súťaž určená?

**Kategória B** má dve kolá: domáce a krajské.

Do kategórie B sa smú zapojiť len tí žiaci, ktorí ešte ani v tomto, ani v nasledujúcom školskom roku nebudú končiť strednú školu.

**Kategória A** má tri kolá: domáce, krajské a celoštátne.

Do kategórie A sa môžu zapojiť všetci žiaci (základných aj) stredných škôl.

Najlepší riešitelia kategórie A majú šancu reprezentovať Slovensko na medzinárodných súťažiach.

### Priebeh súťaže

Za každú úlohu domáceho kola sa dá získať od 0 do 10 bodov. Na základe bodov domáceho kola stanoví Slovenská komisia OI (samostatne pre každú kategóriu) bodovú hranicu potrebnú na postup do **krajského kola**. Očakávame, že táto hranica bude približne rovná **tretine maximálneho počtu bodov**.

V krajskom kole riešitelia riešia štyri teoretické úlohy, ktoré môžu tematicky nadväzovať na úlohy domáceho kola. V kategórii A sú do **celoštátneho kola** pozývaní najúspešnejší riešitelia krajských kôl. Presnejšie, po vyhodnotení krajských kôl prebehne koordinácia bodovacích škál, spoja sa výsledkové listiny do jednej celoštátnej, a do celoštátneho kola sú pozvaní najlepší riešitelia podľa tejto výsledkovej listiny.

V celoštátnom kole účastníci prvý deň riešia tri teoretické úlohy, druhý deň dve praktické úlohy (pri počítaní). Z najlepších riešiteľov tohto kola SK OI na výberovom sústrezení vyberie družstvá pre Medzinárodnú informatickú olympiádu (IOI) a Stredoeurópsku informatickú olympiádu (CEOI).

### Odvzdávanie riešení domáceho kola

V každej kategórii sú úlohy číslo 1 a 2 praktické. Tieto úlohy **je nutné odovzdať prostredníctvom webového rozhrania**, ktoré nájdete na stránkach olympiády (<http://oi.sk/>) najneskôr v októbri 2009.

Úlohy 3 a 4 sú teoretické. Ich riešenia môžete odovzdať **buď klasickou cestou** (odoslaním poštou na adresu príslušnej krajskej komisie), **alebo vo formáte PDF** prostredníctvom vyššie spomínaného webového rozhrania.

### Ako majú vyzeráť riešenia úloh?

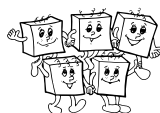
V praktických úlohách je vašou úlohou vytvoriť program v jazyku Pascal, C, alebo C++, ktorý bude riešiť zadanú úlohu. Program musí byť v prvom rade korektný a funkčný, v druhom rade sa snažte aby bol čo najefektívnejší. Presný popis, ako majú vyzeráť riešenia praktických úloh (napr. realizáciu vstupu a výstupu), nájdete na webstránke, kde ich budete odovzdávať.

Odvzdaný program bude automaticky otestovaný na viacerých vopred pripravených testovacích vstupoch. Podľa toho, na koľko z nich dá správnu odpoveď, vám budú pridelené body. Výsledok testovania sa dozviete krátko po odovzdaní. Ak váš program nezíska plný počet bodov, budete ho môcť vylepšiť a odovzdať znova, až do uplynutia termínu na odovzdávanie.

Ak nie je v zadaní povedané ináč, riešenia teoretických úloh musia v prvom rade obsahovať **podrobný slovný popis použitého algoritmu, zdôvodnenie jeho správnosti** a diskusiu o efektívite zvoleného riešenia (t. j. posúdenie časových a pamäťových nárokov programu). Na záver riešenia uveďte program napísaný v jazyku Pascal, C alebo C++. Ak používate v programe netriviálne algoritmy alebo dátové štruktúry (napr. veci z STL v C++), súčasťou popisu algoritmu musí byť dostatočný popis ich implementácie.

### Usporiadateľ súťaže

Olympiádu v informatike (OI) vyhlasuje *Ministerstvo školstva SR* v spolupráci so *Slovenskou informatickou spoločnosťou* a *Slovenskou komisiou Olympiády v informatike*. Súťaž organizuje *Slovenská komisia OI* a v jednotlivých krajoch ju riadia *krajské komisie OI*. Na jednotlivých školách ju zaisťujú učitelia informatiky. Celoštátne kolo OI, tlač materiálov a ich distribúciu po organizačnej stránke zabezpečuje *IUVENTA* v tesnej súčinnosti so Slovenskou komisiou OI.



## Zadania kategórie B

Riešenia kategórie B treba odovzdať najneskôr **30. 11. 2009**.  
V prípade riešení odoslaných poštou rozhoduje pečiatka z pošty.

### B-I-1 Majstrovstvá na rybníku

Ako sa blížila zima, bol na rybníku Čvachtáku čoraz pevnejší ľad. A keď už bol pevný natoľko, že sa ani medveď Mišo nebál po ňom prejsť, zišli sa na brehu zvieratká a začali hútať, ako by sa len zabavili.

„Už viem!“ vykričala liška Eliška. „Spravíme majstrovstvá v krasokorčuľovaní!“

„Ale ja sa neviem korčuľovať...“ namietol had Félix.

„To nevádi,“ vyriešila to Eliška lišiacky, „ty budeš rozhodca!“

#### Súťažná úloha

Zvieratká sa rozdelili na  $S$  súťažiacich a  $R$  rozhodcov. Súťažiaci predviedli svoje jazdy a každý z nich dostal od každého rozhodcu číselnú známku. Výsledné body súťažiaceho zistíme tak, že škrtneme jednu najvyššiu a jednu najnižšiu známku, ktorú dostal, a zvyšných  $R - 2$  známok sčítame. Napíšte program, ktorý načíta všetky známky, vypočíta výsledné body všetkých súťažiacich a vypíše výsledkovú listinu súťaže.

#### Formát vstupu

V prvom riadku vstupu sú dve medzerou oddelené celé čísla  $S$  a  $R$  – počet súťažiacich a počet rozhodcov. Môžete predpokladať, že  $3 \leq S, R$  a že  $S \cdot R \leq 1\,000\,000$ .

Nasleduje  $2S$  riadkov, vždy dva riadky popisujú jedného súťažiaceho. V prvom z nich je meno súťažiaceho. Meno je reťazec tvorený medzi 1 a 10 malými písmenami anglickej abecedy. V druhom riadku je vždy zoznam  $R$  medzerami oddelených celých čísel, predstavujúcich známky, ktoré tento súťažiaci dostal od rozhodcov. Známky sú z rozsahu od 0 po 6 000, vrátane.

V niektorých 6 testovacích vstupoch bude platiť  $S \leq 1\,000$ . V niektorých 4 testovacích vstupoch budú mať všetci súťažiaci navzájom rôzne výsledné body.

#### Formát výstupu

Váš program má vypísať  $S$  riadkov – jeden pre každého súťažiaceho. Každý riadok má mať podobu „B: meno“, kde meno je meno súťažiaceho a B sú body, ktoré získal. Riadky usporiadajte podľa počtu bodov zostupne. Súťažiacich s rovnakým skóre vypíšte usporiadaných podľa abecedy vzostupne (napr. `baran < had < hadica`).

#### Príklad

vstup

```
4 5
zajko
8 7 6 5 4
jezko
4 4 4 4 4
zabka
7 13 7 4 4
medvedik
23 19 47 11 48
```

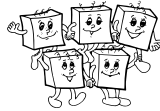
výstup

```
89: medvedik
18: zabka
18: zajko
12: jezko
```

Všimnite si, že zabka a zajko majú rovnaké skóre, preto sú utriedení podľa abecedy.

#### Odvzdávanie riešení

Toto je praktická úloha. Odovzdávate **len funkčný, odladený program** prostredníctvom webového rozhrania.



## B-I-2 Cesta

Firma TravelEarth plánuje v novom roku do svojich GPS navigačných zariadení pridať novú funkciu. Tá by mala pomôcť vodičom presnejšie určiť dĺžku jazdy autom pozdĺž zvolenej trasy. Takýto čas dnes TravelEarth odhaduje iba na základe vzdialeností. Lenže v skutočnosti sa popri ceste nachádzajú dopravné značky, ktoré na rôznych úsekoch povoľujú rôznu maximálnu rýchlosť.

### Súťažná úloha

V našej úlohe budeme uvažovať len jednu konkrétnu cestu. Keď na nej vyznačíme miesta, kde sa mení maximálna povolená rýchlosť, rozdelíme ju tak na  $N$  úsekov. Tieto úseky si očísľujeme od 1 do  $N$ . Dĺžku úseku  $i$  označíme  $d_i$  a maximálnu povolenú rýchlosť na ňom označíme  $v_i$ .

Vašou úlohou je napísať program, ktorý načíta a spracuje popis cesty a následne bude odpovedať na otázky tvaru: „Ako dlho nám bude trvať presun, ak začíname na  $x$ -tom a končíme na  $y$ -tom kilometri cesty?“ (Samozrejme za predpokladu, že v každom úseku ideme presne jeho maximálnou povolenou rýchlosťou.)

### Formát vstupu

Prvý riadok vstupu obsahuje jedno celé číslo  $N$  ( $1 \leq N \leq 200\,000$ ) – počet úsekov na našej ceste. Druhý riadok obsahuje  $N$  medzerami oddelených celých čísel  $d_1, \dots, d_N$  ( $1 \leq d_i \leq 10\,000$ ) – dĺžky jednotlivých úsekov v kilometroch. Tretí riadok obsahuje  $N$  medzerami oddelených celých čísel  $v_1, \dots, v_N$  ( $1 \leq v_i \leq 500$ ) – maximálne povolené rýchlosti v kilometroch za hodinu.

Vo štvrtom riadku nasleduje celé číslo  $Q$  ( $1 \leq Q \leq 100\,000$ ) – počet otázok. Každý z nasledujúcich  $Q$  riadkov obsahuje dve medzerou oddelené celé čísla  $x$  a  $y$  predstavujúce jednu otázku ( $0 \leq x < y \leq D$ , kde  $D$  je súčet všetkých  $d_i$ , t. j. celková dĺžka cesty).

V 7 z 10 testovacích vstupov bude  $D \leq 16\,000\,000$ . V 5 z týchto vstupov bude tiež  $N \leq 10\,000$  a  $Q \leq 20\,000$ .

### Formát výstupu

Pre každú otázku vypíšte jeden riadok – najkratší čas v hodinách, za ktorý sa pri dodržaní predpisov vieme dostať z  $x$ -tého kilometra cesty na  $y$ -ty.

Na ukladanie reálnych čísel odporúčame použiť dátový typ `double`. Na výpis výsledku môžete použiť príkaz `writeln(vysledok:0:7)`; ak programujete v Pasmale, resp. `printf("%.7f\n", vysledok)`; v C/C++. Výsledok vypíšte na aspoň 7 desatinných miest. Riešenia, ktoré sa od správneho budú líšiť len drobnou zaokrúhľovacou chybou, uznáme za správne.

### Príklad

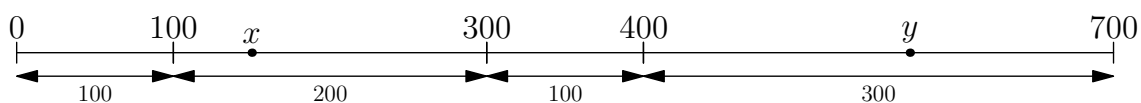
vstup

```
4
100 200 100 300
80 90 50 120
3
300 400
0 700
150 570
```

výstup

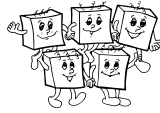
```
2.0000000
7.9722222
5.0833333
```

*Cesta zo vstupu je znázornená na obrázku. Body  $x$  a  $y$  zodpovedajú tretej otázke. Pri ceste z  $x$  do  $y$  pôjdeme prvých 150 km rýchlosťou 90 km/h, potom 100 km rýchlosťou 50 km/h, a nakoniec 170 km rýchlosťou 120 km/h.*



### Odvzdávanie riešení

Toto je praktická úloha. Odovzdávate **len funkčný, odladený program** prostredníctvom webového rozhrania.

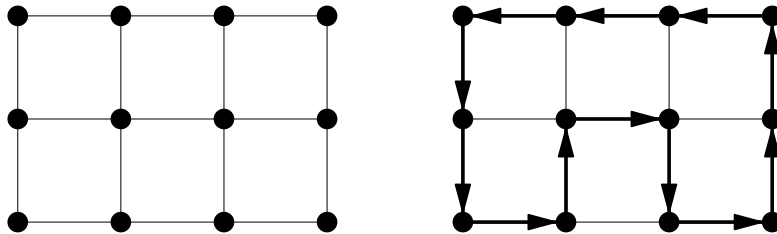


### B-I-3 Squaplex

Squaplex je hlavolam, ktorý sa hrá na štvorcovanom papieri v tvare obdĺžnika, obsahujúcom  $R \times S$  mrežových bodov. Úlohou riešiteľa je položiť hrot ceruzky do ktoréhokoľvek mrežového bodu a nakresliť ťah splňajúci nasledujúce podmienky:

- Ceruzkou sa môžeme pohybovať len po čiarach hracieho plánu.
- Ťah musí začínať aj končiť v tom istom mrežovom bode.
- Cez každý mrežový bod na hracej ploche musíme ceruzkou prejsť práve raz. (Výnimkou je začiatočný bod, ktorý navštívime dvakrát.)

Na obrázku je hrací plán veľkosti  $3 \times 4$  (vľavo) a jedno možné riešenie preň (vpravo).



#### Súťažná úloha

- (5 bodov) Zistite a dokážte, pre ktoré hodnoty  $R$  a  $S$  má tento hlavolam riešenie.
- (5 bodov) Napíšte program, ktorý pre zadané hodnoty  $R$  a  $S$  navrhne jedno riešenie, ak existuje.

#### Formát vstupu

Vstup obsahuje dve kladné celé čísla  $R$  a  $S$ , oddelené medzerou. Môžete predpokladať, že aspoň jedno z čísel  $R$  a  $S$  je väčšie ako 1.

#### Formát výstupu

Ak neexistuje pre dané  $R$  a  $S$  riešenie, vypíšte jeden riadok so správou „Nema riešenie.“.

V opačnom prípade vypíšte popis jedného riešenia, ktoré začína aj končí v ľavom hornom rohu. Riešenie môžeme popísať ako postupnosť znakov H (hore), D (dole), L (doľava) a P (doprava).

#### Príklad

vstup

3 4

výstup

DDPHDPHLLL

*Táto postupnosť pohybov zodpovedá riešeniu na obrázku v zadání.*

#### Odvzdávanie riešení

Toto je teoretická úloha. Riešenie, splňajúce požiadavky uvedené v pravidlách, buď pošlite poštou na adresu vašej krajskej komisie, alebo odovzdajte vo formáte PDF prostredníctvom webového rozhrania.

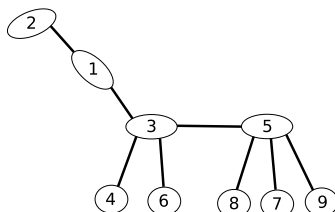


### B-I-4 Gaštanová žirafa

Zuzkino najobľúbenejšie ročné obdobie je jeseň. Vždy sa teší na deň, kedy dozrejú prvé gaštany, a ona si z nich bude môcť stavať rôzne zvieratká.

Určite ste už aj vy také zvieratko videli. Vyrába sa veľmi ľahko – zoberieme zápalku, oba jej konce zabodneme do dvoch gaštanov, a už máme prvú nožičku.

Zuzka si dnes nazbierala  $N$  gaštanov. Najskôr si na ne fixkou napísala čísla od 1 po  $N$ , lebo sa chcela hrať s mladším bratom Tomáškom na lotériu. Potom ale všetky gaštany zobrala a postavila z nich *veľanohú žirafu*.



Na obrázku vidíme príklad veľanohej žirafy. Každá veľanohá žirafa sa skladá z nasledujúcich gaštanov:

- 4 gaštany predstavujúce hlavu, krk, trup a zadok žirafy.
- aspoň 2 gaštany predstavujúce predné kopytá
- aspoň 2 gaštany predstavujúce zadné kopytá

Zápalkami sú prepojené nasledujúce dvojice gaštanov: hlava-krk, krk-trup, trup-zadok, trup-každé predné kopyto a zadok-každé zadné kopyto.

Zuzka vymyslela pre Tomáška nasledujúcu hru: Na začiatku Zuzka oznámi Tomáškovi hodnotu  $N$ . Tomáškovým cieľom je o každom gaštane od 1 do  $N$  zistiť, ktorú časť žirafy predstavuje. Tomáško však žirafu nevidí, lebo ju Zuzka pred ním schovala. Môže sa jej len pýtať otázky tvaru „sú gaštany  $a$  a  $b$  spojené zápalkou?“

#### Príklad hry

Z: Počet gaštanov tvoriacich moju žirafu je 9.

T: Sú gaštany 1 a 2 spojené? Z: Áno.

T: Sú gaštany 1 a 3 spojené? Z: Áno.

T: Sú gaštany 3 a 4 spojené? Z: Áno.

T: Sú gaštany 3 a 5 spojené? Z: Áno.

T: Sú gaštany 3 a 6 spojené? Z: Áno.

T: Sú gaštany 3 a 7 spojené? Z: Nie.

T: Sú gaštany 5 a 8 spojené? Z: Áno.

T: Sú gaštany 5 a 9 spojené? Z: Áno.

T: Hlava je gaštan 2. Krk je gaštan 1. Trup je gaštan 3. Zadok je gaštan 5.

Predné kopytá sú gaštany 4 a 6. Zadné kopytá sú gaštany 7, 8 a 9.

(Zuzkina žirafa v tomto príklade je tá z obrázku.)

#### Súťažná úloha

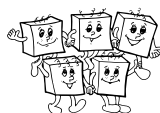
Navrhňte čo najlepšiu stratégiu pre túto hru. Presnejšie, nájdite čo najefektívnejší algoritmus, ktorý bude túto hru hrať namiesto Tomáška a vždy vyhrá. Algoritmus stačí v riešení uviesť ako pseudokód (slovný popis), nie je potrebné implementovať ho v programovacom jazyku.

Na zisk 10 bodov potrebujete najst algoritmus, ktorého časová zložitosť bude lineárna od  $N$ . Za 7 bodov bude riešenie, ktorého časová zložitosť bude od  $N$  závisieť kvadraticky.

Pri odhade časovej zložitosti predpokladajte, že Zuzka dokáže na každú otázku odpovedať okamžite. Uvedomte si, že 10-bodové riešenie si ani zďaleka nemôže dovoliť opýtať sa Zuzky na všetky dvojice gaštanov.

#### Odovzdávanie riešení

Toto je teoretická úloha. Riešenie, spĺňajúce požiadavky uvedené v pravidlách, buď pošlite poštou na adresu vašej krajskej komisie, alebo odovzdajte vo formáte PDF prostredníctvom webového rozhrania.



## Zadania kategórie A

Riešenia kategórie A treba odovzdať najneskôr **16. 11. 2009**.  
V prípade riešení odoslaných poštou rozhoduje pečiatka z pošty.

### A-I-1 Maliar Bonifác

Mestský úrad v Kocúrkove vypísal nedávno výberové konanie na veľmi zodpovednú a dôležitú úlohu: maľovanie chodníka pred mestským úradom. Výberové konanie vyhral maliar Bonifác (jediný uchádzač (a starostov brat)).

Ako to už chodí, len čo Bonifác podpísal zmluvu, začali mu z mestského úradu prichádzať príkazy: Tento kus chodníka zelenou, tento ružovou, potom to skoro celé pretrieť na bielo. . . Veru tak. Netrvalo dlho a Bonifác si všimol, že niektoré príkazy sa prekrývajú. A keď si spomenul, že ho zmluva zaväzuje všetky príkazy, v tom poradí v akom ich dostal, vykonať, začali ho obchádzať mdloby.

No zrazu prišiel na myšlienku vskutku geniálnu. Keby vedel, ako má chodník vyzeráť na konci, po vykonaní všetkých príkazov, mohol by ho tak vymaľovať rovno a potom sa tváriť, že on predsa všetky príkazy dodržal. A hlavne potom mestskému úradu všetko vyúčtuje podľa pôvodných príkazov a pekne sa na tom nabalí.

#### Súťažná úloha

Chodník pred mestským úradom má  $K$  kocúrkovských krokov. Ten jeho koniec, ktorý je bližšie ku potravinám, bude mať súradnicu 0, opačný koniec bude mať súradnicu  $K$ . V súčasnosti má celý chodník asfaltovo modrú farbu. Bonifác má  $F$  iných farieb, očíslovaných od 1 po  $F$ . Postupne mu prišlo  $N$  príkazov, každý z nich je tvaru „ $a_i b_i f_i$ “, kde  $a_i$  a  $b_i$  sú súradnice začiatku a konca úseku a  $f_i$  je farba, ktorou ho má ofarbiť. Na ofarbenie metra chodníka potrebuje Bonifác liter farby. Pre každú z farieb spočítajte, koľko litrov jej Bonifác bude potrebovať.

#### Formát vstupu

V prvom riadku vstupu sú tri medzerami oddelené celé čísla  $N$ ,  $F$  a  $K$  ( $1 \leq N, F \leq 100\,000$ ,  $1 \leq K \leq 1\,000\,000\,000$ ). Nasleduje  $N$  riadkov, z ktorých každý popisuje jeden príkaz, a to v poradí, v akom ich Bonifác dostal. Presnejšie,  $i$ -ty z týchto riadkov obsahuje tri medzerami oddelené celé čísla  $a_i$ ,  $b_i$  a  $f_i$  ( $0 \leq a_i < b_i \leq K$ ,  $1 \leq f_i \leq F$ ).

Pre 8 z 10 testovacích vstupov bude navyše platiť  $K \leq 1\,000\,000$ . Pre 6 z týchto 8 bude navyše aj  $N \leq 10\,000$ . Pre 3 z týchto 6 bude  $N, K \leq 1\,000$ .

#### Formát výstupu

Pre každú z  $F$  farieb vypíšte jeden riadok a v ňom jedno celé číslo – počet litrov tejto farby, ktoré budeme potrebovať.

#### Príklad

vstup

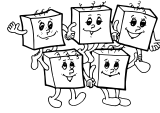
```
4 5 7
1 5 1
2 4 3
4 6 4
3 6 2
```

výstup

```
1
3
1
0
0
```

#### Odvzdávanie riešení

Toto je praktická úloha. Odovzdávate **len funkčný, odladený program** prostredníctvom webového rozhrania.



## A-I-2 Čokoláda

Marienka bude mať čoskoro narodeniny. Jej braček Janko dlho nevedel, čo jej darovať – až kým vo svojej tajnej skrýši na povale nenarazil na zvyšky čokolády, ktorú si tam kedysi ukryl. Pravda, myši už boli vybrať svoju daň, ale aj tak jej ešte zostalo dosť. Deravé časti oláme, aby mu ostala pekná štvorcová tabuľka, a tú pekne zabalí. A zvyšok samozrejme zje.

### Súťažná úloha

Daný je pôvodný počet riadkov  $R$  a stĺpcov  $S$ , ktoré čokoláda kedysi mala, a matica  $R \times S$  núl a jednotiek udávajúca, ktoré políčka z nej zostali celé. Zistíte, koľkými spôsobmi môže Janko uskutočniť svoj plán. Inými slovami, spočítajte, koľkými spôsobmi sa dá na zvyšku čokolády vyznačiť štvorec bez dier. Všetky hrany štvorca musia samozrejme ležať na hranách políčok. Rovnako veľké štvorce na rôznych súradniciach považujeme za rôzne.

### Formát vstupu

V prvom riadku vstupu sú dve medzerami oddelené celé čísla  $R$  a  $S$  ( $1 \leq R, S \leq 2500$ ). Nasleduje  $R$  riadkov, v  $r$ -tom z nich je  $S$  medzerami oddelených celých čísel  $a_{r,1}, \dots, a_{r,s}$ . Ak je políčko  $(r, s)$  celé, je  $a_{r,s} = 1$ , inak  $a_{r,s} = 0$ .

Pre 7 z 10 testovacích vstupov bude platiť  $R \leq 500$  a  $S \leq 2500$ . Pre 5 z týchto 7 bude  $R, S \leq 500$ , a pre 3 z týchto 5 bude  $R, S \leq 20$ .

### Formát výstupu

Vypíšte jeden riadok a v ňom jedno celé číslo – hľadaný počet štvorcov.

### Príklad

vstup

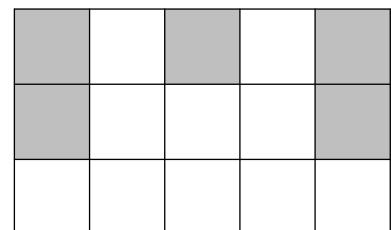
```
3 5
0 1 0 1 0
0 1 1 1 0
1 1 1 1 1
```

výstup

```
12
```

Na obrázku vpravo je čokoláda popísaná vstupom. Sivou farbou sú políčka, ktoré chýbajú.

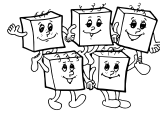
Štvorec  $1 \times 1$  sa na nej dá vyznačiť desiatimi spôsobmi a štvorec  $2 \times 2$  dvoma, to je dokopy  $10 + 2 = 12$  spôsobov.



### Odvzdávanie riešení

Toto je praktická úloha. Odovzdávate **len funkčný, odladený program** prostredníctvom webového rozhrania.



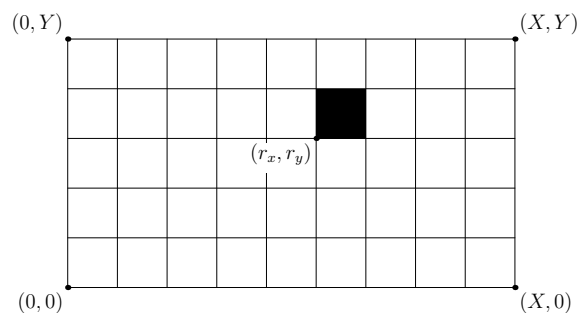


### A-I-3 Koláč

Zlá ježibaba drží v kletke Janka a Marienku a snaží sa ich vykrmiť. Práve pre nich upiekla za plech ježibabieho koláča. Koláč má tvar obdĺžnika. Celý je odpudivý a Jankovi s Marienkou sa doň príliš nechce. A keby to nestačilo, namiesto príslovečnej čerešničky je koláč ozdobený ohavnou pečenou ropuchou.

Keďže čokoliek je lepšie ako musieť zjesť túto ropuchu, rozhodli sa Janko s Marienkou, že si z jedenia koláča spravia hru. Marienka na ňom lyžičkou naznačila čiary, čím ho rozdelila na  $X \times Y$  rovnakých štvorcov. Celá ropucha sedí na jednom z týchto štvorcov.

Teraz budú striedavo ťahať. Hráč, ktorý je na ťahu, si vyberie niektorú z vyznačených čiar, a pozdĺž nej koláč rozreže na dve obdĺžnikové časti. Následne tú časť, kde nie je ropucha, zje. Samozrejme, ten, kto príde na ťah v okamihu, keď už zostal len štvorec s ropuchou, prehral a musí ju zjesť. Ako prvá ťahá Marienka.



#### Súťažná úloha

Dané sú rozmery koláča – kladné celé čísla  $X, Y$ . Tiež sú dané súradnice  $r_x, r_y$  ( $0 \leq r_x < X, 0 \leq r_y < Y$ ) ľavého dolného rohu štvorca, v ktorom je ropucha.

- (2 body) Rozhodnite, kto vyhrá hru pre situáciu, ktorá je na obrázku – teda pre  $(X, Y) = (9, 5)$  a  $(r_x, r_y) = (5, 3)$  – a popíšte jednu možnú stratégiu, ktorá mu zabezpečí výhru.
- (8 bodov) Napište čo najefektívnejší program, ktorý pre dané hodnoty  $X, Y, r_x$  a  $r_y$  zistí, ktoré z detí hru vyhrá, ak budú obe hrať optimálne.

#### Príklady

vstup

8 1 1 0



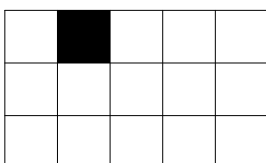
output

Vyhra Marienka.

V prvom ťahu spraví rez po priamke  $x = 3$ . Zostane ropucha a okolo nej z každej strany jeden štvorec. Janko zje jeden z nich, Marienka druhý, a Jankovi zostane ropucha.

vstup

5 3 1 2

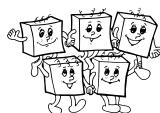


output

Vyhra Janko.

#### Odvzdávanie riešení

Toto je teoretická úloha. Riešenie, spĺňajúce požiadavky uvedené v pravidlách, buď pošlite poštou na adresu vašej krajskej komisie, alebo odovzdajte vo formáte PDF prostredníctvom webového rozhrania.



### A-I-4 Počítač s gumenou rúrou

V tomto ročníku olympiády sa budeme v teoretickej úlohe stretávať so špeciálnym počítačom zvaným Kvak. V študijnom texte uvedenom za zadaním tejto úlohy je popísané, ako tento počítač funguje.

#### Súťažná úloha

a) (3 body)

V rúre je jedno číslo. Napíšte program pre Kvak, ktorý vypíše 1, ak je to prvočíslo, a 0 inak.

Plný počet bodov dostanete za ľubovoľné riešenie, ktoré bude mať menej ako 100 príkazov a na ľubovoľnom vstupe spraví menej ako 10 000 krokov.

b) (4 body)

V rúre je postupnosť *kladných* čísel. Dĺžka postupnosti je menšia ako 65 000. Napíšte program pre Kvak, ktorý túto dĺžku spočíta a vypíše. Plný počet bodov dostanete za riešenie, ktoré bude mať časovú zložitosť lineárnu od dĺžky vstupnej postupnosti.

c) (3 body)

Máme pokazenú verziu počítača Kvak, ktorá sa od obvyčajnej líši tým, že inštrukciu `put` už zvládne vykonať len desaťkrát a potom sa definitívne zasekne. Ľubovoľné iné inštrukcie zvláda vykonávať bez problémov.

V rúre je neprázdna postupnosť čísel, ktorá môže byť ľubovoľne dlhá. Dá sa napísať program pre pokazený Kvak, ktorý nájde a vypíše jej maximum? Ak áno, napíšte ho, ak nie, dokažte, že to nejde.

#### Interpreter

Aby ste si mohli otestovať svoje riešenia, na webstránke olympiády máte k dispozícii interpreter programov pre Kvak. Nájdete ho na adrese <http://oi.sk/kvak/>. Môžete ho použiť buď online, alebo si stiahnuť jeho zdrojový kód (v C++) a doma si ho skompilovať.

#### Odvzdávanie riešení

Toto je teoretická úloha. Riešenie, spĺňajúce požiadavky uvedené v pravidlách, buď pošlite poštou na adresu vašej krajskej komisie, alebo odovzdajte vo formáte PDF prostredníctvom webového rozhrania.

### Študijný text

Vedci z Kolégia Skúmania Potrubí (KSP) nedávno vyvinuli nový počítač, zvaný Kvak. Kvak má jedinú dátovú štruktúru: jednosmernú gumenú rúru.

Jediný dátový typ, ktorý Kvak pozná, sa volá `number`, a je to celé číslo z rozsahu od 0 po 65 535, vrátane.<sup>1</sup> Všetky matematické operácie počíta Kvak modulo 65 536. Teda napríklad hodnota výrazu  $65530 + 10$  je 4.

Kvak má 26 premenných, ktoré voláme registre. Tie sú označené písmenami `a` až `z` a v každom z nich môže byť uložená jedna hodnota typu `number`. Na začiatku výpočtu sú vo všetkých registroch nuly.

Okrem registrov má Kvak, ako sme už spomínali, jednu jednosmernú gumenú rúru. S tou vie robiť dve operácie:

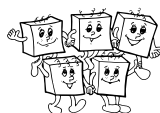
- vložiť do nej číslo z registra `X` príkazom `put X`,
- z opačného konca rúry vybrať číslo a uložiť ho do registra `X` príkazom `get X`.

Čísla sa v rúre samozrejme nemôžu predbiehať, Kvak ich teda bude vyberať v tom istom poradí, v akom ich tam vložil. Rúra je gumená, takže sa do nej čísel zmestí ľubovoľne veľa. Ak nie je povedané ináč, je na začiatku výpočtu rúra prázdna.

Okrem rúry má Kvak ešte jeden kotúč žltej pásky, na ktorú môže písať svoj výstup.

V nasledujúcej tabuľke sú zhrnuté všetky príkazy, ktoré Kvak pozná, a ktoré vy môžete použiť, keď ho budete programovať.

<sup>1</sup>  $65\,535 = 2^{16} - 1$ , typ `number` je teda 16-bitové celé číslo bez znamienka (to isté ako `word` v Pascale, `unsigned short` v C/C++).



príkaz	účinnok príkazu
get X	Kvak vyberie číslo z rúry a uloží ho do registra X.
put X	Kvak vloží do rúry číslo z registra X.
put číslo	Kvak vloží dané číslo do rúry.
print	Kvak vyberie číslo z rúry a vypíše ho na výstup.
add	sčítanie: Kvak vyberie dve čísla z rúry a vloží tam ich súčet.
sub	odčítanie: Kvak vyberie dve čísla z rúry a vloží tam ich rozdiel (prvé mínus druhé).
mul	násobenie: Kvak vyberie dve čísla z rúry a vloží tam ich súčin.
div	delenie: Kvak vyberie dve čísla z rúry a vloží tam celú časť ich podielu (prvé lomeno druhé).
mod	zvyšok: Kvak vyberie dve čísla z rúry a vloží tam zvyšok, ktorý dá prvé z nich po delení druhým.
label L	návestie: Toto miesto v programe dostane meno L (kde L môže byť ľubovoľný reťazec).
jump L	skok: Kvak bude pokračovať vo vykonávaní programu od miesta, ktoré sa volá L.
jz X L	skok ak nula: Ak je v registri X nula, Kvak vykoná príkaz jump L.
jeq X Y L	skok ak sa rovnajú: Ak je v registroch X a Y to isté, Kvak vykoná príkaz jump L.
jgt X Y L	skok ak je väčšie: Ak je v registri X väčšia hodnota ako v Y, Kvak vykoná príkaz jump L.
jempty L	skok ak je rúra prázdna: Ak v rúre nie sú žiadne čísla, Kvak vykoná príkaz jump L.
stop	koniec: Kvak prestane vykonávať program.

Ak sa stane, že pri pokuse vybrať z rúry je rúra prázdna, nastane chyba. Rovnako nastane chyba, ak sa pokúsime deliť nulou, počítať zvyšok po delení nulou, alebo skočiť na neexistujúce miesto. Ak sa program dostane na koniec, Kvak korektne skončí (ako keby na konci programu bola ešte inštrukcia `stop`.)

Pre stručnosť môžeme písať viac príkazov do jedného riadku, v takomto prípade ich od seba treba oddeliť bodkočiarkou.

### Príklad 1

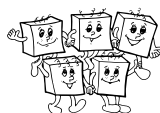
Nasledujúci program spočíta a vypíše súčet čísel od 1 do 20.

```
put 20
put 0
label start
get n
jz n end
put n ; put n ; put 1
add
sub
get x ; put x
jump start
label end
print
```

Vždy, keď sa Kvak pri vykonávaní programu dostane k tretiemu riadku (`label start`), budú v rúre práve dve čísla. Keď prvé z nich označíme  $n$ , hodnota druhého bude súčet  $s = (n + 1) + \dots + 20$ . Následne načítame  $n$  do registra  $n$ . Ak je  $n = 0$ , máme v rúre hľadaný súčet, môžeme ho vypísať a skončiť. V opačnom prípade chceme spraviť dve veci: Prirábať toto  $n$  k doteraz získanému súčtu, a následne zmenšiť  $n$  o jedna. Po vykonaní riadku šesť (tri príkazy `put`) máme v rúre postupne čísla:  $s, n, n, 1$ . Príkaz `add` sčíta prvé dve, po jeho vykonaní je v rúre:  $n, 1, n + s$ . Po vykonaní ďalšieho príkazu `sub` máme v rúre hodnoty  $n + s$  a  $n - 1$ . To už je skoro to, čo treba, len sú v opačnom poradí. Preto jednu z nich načítame do  $x$  a znovu vložíme.

### Príklad 2

V rúre je neprázdna postupnosť čísel. Napíšeme program, ktorý spočíta a vypíše jej súčet. (Presnejšie, jeho zvyšok po delení 65 536.)



Dokola budeme opakovať nasledujúci postup: Zistíme, či sú v rúre aspoň dve čísla. Ak áno, tie dve, ktoré sú práve na začiatku, sčítame. Ak už zostalo len jedno, je zjavne súčtom všetkých pôvodných čísel. V programe pre Kvak môžeme túto myšlienku implementovať napríklad nasledovne:

```
label cyklus
get a
jempty koniec
put a
add
jump cyklus

label koniec
put a
print
```

Na začiatku každej iterácie načítame do registra a číslo z rúry. Ak je tá v tomto okamihu prázdna, máme v registri a hľadaný súčet, stačí ho už len vypísať. Ak nie, číslo z registra a vrátime do rúry. V tomto okamihu sú v rúre aspoň dve čísla, a teda môžeme bez obáv zavolať inštrukciu add.

Časová zložitosť tohto riešenia je lineárna od počtu čísel, ktoré boli na začiatku v rúre. Totiž každá iterácia cyklu trvá len konštantne veľa krokov a zmenší nám o jedno počet čísel v rúre.

---

#### Adresy krajských komisií OI

##### *Banskobystrický kraj*

PaedDr. Miloslava Sudolská, KI FPV UMB, Tajovského 40, 974 01 B. Bystrica

##### *Bratislavský kraj*

RNDr. Eva Hanulová, Gymnázium Jura Hronca, Novohradská 3, 821 09 Bratislava

##### *Košický kraj*

RNDr. Rastislav Krivoš-Belluš, Ústav informatiky PF UPJŠ, Jesenná 5, 041 54 Košice

##### *Nitriansky kraj*

prof. Ing. Veronika Stoffová, CSc., KI PF UJS, Roľníckej školy 1519, 945 01 Komárno

##### *Prešovský kraj*

Mgr. Mária Majherová, Ph.D., Katedra matematiky, FHPV PU, Ul. 17 novembra č. 1, 081 16 Prešov

##### *Trenčiansky kraj*

Ing. Andrea Julény, Katedra informatiky, Fakulta mechatroniky TnUAD, Študentská 2, 911 50 Trenčín

##### *Trnavský kraj*

Mgr. Blanka Thomková, Gym. Jána Hollého, Na hlinách 30, 917 01 Trnava

##### *Žilinský kraj*

RNDr. Peter Varša, Ph.D., KI FRI ŽU, Moyzesova 20, 010 26 Žilina

---

### SLOVENSKÁ KOMISIA OLYMPIÁDY V INFORMATIKE DVADSIATY PIATY ROČNÍK OLYMPIÁDY V INFORMATIKE

Vydala IUVENTA s finančnou podporou Ministerstva školstva SR

Náklad: 400 výtlačkov

Zodpovedný redaktor: Michal Forišek

Sadzba programom L<sup>A</sup>T<sub>E</sub>X

© Slovenská komisia Olympiády v informatike, 2009