



Informácie a pravidlá

Pre koho je súťaž určená?

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

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

Odvzdávanie riešení domáceho kola

Riešitelia domáceho kola odovzdávajú riešenia sami, v elektronickej podobe, a to priamo na stránke olympiády: <http://oi.sk/>. Odovzdávanie riešení bude spustené niekedy v septembri.

Riešenia kategórie A je potrebné odovzdať najneskôr **15. novembra 2016**.

Riešenia kategórie B je potrebné odovzdať najneskôr **1. decembra 2016**.

Priebeh súťaže

Za každú úlohu domáceho kola sa dá získať od 0 do 10 bodov. Na základe bodov domáceho kola stanoví Slovenská komisia OI (SK OI) 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 B súťaž týmto kolom končí.

V kategórii A je približne najlepších 30 riešiteľov krajského kola (podľa počtu bodov, bez ohľadu na kraj, v ktorom súťažili) pozvaných do **celoštátneho kola**. V celoštátnom kole účastníci prvý deň riešia teoretické a druhý deň praktické úlohy. Najlepší riešitelia sú vyhlásení za víťazov. Približne desať najlepších riešiteľov následne SK OI pozve na týždňové výberové sústredenie. Podľa jeho výsledkov SK OI vyberie družstvá pre Medzinárodnú olympiádu v informatike (IOI) a Stredoeurópsku olympiádu v informatike (CEOI).

Ako majú vyzeráť riešenia úloh?

V praktických úlohách je vašou úlohou vytvoriť program, ktorý bude riešiť zadanú úlohu. Program musí byť v prvom rade korektný a funkčný, v druhom rade sa snažte aby bol čo najefektívnejší.

V kategórii B môžete použiť ľubovoľný programovací jazyk.

V kategórii A musíte riešenia praktických úloh písať v jednom z podporovaných jazykov (napr. C++, Pascal alebo Java). Odovzdaný 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.

Presný popis, ako majú vyzeráť riešenia praktických úloh (napr. realizáciu vstupu a výstupu), nájdete na webstránke, kde ich budete odovzdávať.

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 efektivite zvoleného riešenia (t. j. posúdenie časových a pamäťových nárokov programu). Na záver riešenia uveďte program. Ak používate v programe netriviálne algoritmy alebo dátové štruktúry (napr. rôzne súčasti 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* (odborným garantom súťaže) 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.



B-I-1 Pokémoni

Počas leta vyšla nová hra využívajúca rozšírenú realitu – Pokémon Go. Má jednoduchý princíp: stačí zobrať do ruky mobil a ísť sa poprechádzať po okolí. Občas narazíte na nejakého pokémona (magické zvieratko), ktorého môžete chytiť tak, že po ňom hodíte prázdnu pokéloptu (zariadenie na uskladnenie jedného pokémona). Inokedy narazíte na pokéstop, v ktorom zadarmo získate pokélopty na ďalší lov pokémonov.

Peto každý deň pobehuje s mobilom a chytá pokémonov. Rôzni pokémoni zväčša majú rôznu silu. No a Peťovi sa občas stane, že stretne naozaj silného pokémona, ale keďže práve nemá prázdnu pokéloptu, nevie ho chytiť. Každý večer preto rozmýšľa, akých najsilnejších pokémonov vedel chytiť, keby čo najšikovnejšie použil pokélopty.

Súťažná úloha

Na vstupe dostanete záznam stretnutí, ktoré v daný deň Peťo zažil: pozorovania pokémonov a návštevy pokéstopov. Na začiatku dňa má Peťo m prázdnych pokélôpt. V každom pokéstope si Peťo vie nabráť ľubovoľne veľa prázdnych pokélôpt, nikdy však nesmie mať pri sebe viac ako m prázdnych pokélôpt naraz. Keď Peťo stretne pokémona, musí sa rozhodnúť, či ho chce chytiť. Ak ho chce, hodí po ňom jednu prázdnu pokéloptu a zaručene ho do nej chytí. Ak sa rozhodne pokémona nechytiť, ten zmizne. (Peťo sa teda k nemu už neskôr nemôže vrátiť.) Vašou úlohou je zistiť, aký je najväčší možný súčet síl pokémonov, ktorých v ten deň Peťo vedel pochytať.

Formát vstupu a výstupu

Dostanete od nás 5 vstupných súborov, označených **1.txt** až **5.txt**. Každý z nich má nasledujúci formát:

V prvom riadku sú dve medzerou oddelené čísla n a m . Číslo n udáva počet udalostí, ktoré sa v ten deň Peťovi prihodia, číslo m sme popísali vyššie.

V druhom riadku je n medzerami oddelených čísel x_1 až x_n popisujúcich jednotlivé udalosti v poradí, v akom sa Peťovi stali. Ak je číslo x_i väčšie ako 0, znamená to, že Peťo narazil na pokémona so silou x_i , ktorého môže chytiť. Ak je $x_i = -1$, Peťo navštívil pokéstop a môže si doplniť zásobu pokélôpt.

Na výstup vypíšete jediné číslo – najväčší súčet síl pokémonov, ktorých vie v ten deň Peťo chytiť.

Veľkosti vstupov

V jednotlivých vstupoch má premenná n postupne hodnoty 100, 1 000, 10 000, 500 000 a 1 000 000 a premenná m postupne hodnoty 10, 2, 500, 63 000 a 85 000. Navyše vo vstupoch **1.txt**, **2.txt** a **4.txt** sú všetky čísla x_i veľké nanejvýš 10^6 . Vo zvyšných vstupoch môžu mať veľkosť až 10^9 .

Keďže výsledné číslo môže byť naozaj veľké a nemusí sa zmestiť do klasickej 32-bitovej premennej, odporúčame používať 64-bitové premenné, akými sú `long long` v C++ alebo `int64` v Pascale.

Odvodzovanie riešení

Toto je praktická úloha. Napíšte v ľubovoľnom programovacom jazyku program, ktorý ju rieši.

Zo stránky <http://oi.sk/> stiahnite ZIP archív obsahujúci 5 testovacích vstupov, nazvaných **1.txt** až **5.txt**.

Vyrobte k čo najviac vstupom správne výstupy a uložte ich do súborov **sol1.txt** až **sol5.txt**.

Odvzdajte ZIP archív obsahujúci zdrojový kód vášho programu a tieto výstupné súbory.

Za každý správny výstupný súbor získate 2 body.

Príklad

vstup

```
9 2
9 -1 3 2 4 -1 9 -1 1
```

výstup

```
26
```

Najlepšie riešenie je chytiť pokémonov so silami 9, 3, 4, 9 a 1. Jeden možný postup: Na začiatku dňa má Peťo dve pokélopty. Do jednej chytí prvého pokémona (so silou 9). Následne nájde pokéstop a doplní svoju zásobu lôpt. Potom chytí pokémona (so silou 3). Ďalšieho pokémona (so silou 2) nechá ujsť, poslednú loptu si ušetrí na nasledujúceho (so silou 4). Na ďalšom pokéstope zoberie jednu loptu, do nej potom chytí pokémona (so silou 9). Na poslednom pokéstope si vezme trebárs aj dve lopty. Nakoniec do jednej chytí pokémona (so silou 1).



B-I-2 Obracanie poľa

Máme pole celých čísel. Políčka nášho poľa sú očíslované od 0 po $n - 1$, pre nejaké n . Dĺžka nášho poľa (číslo n) je násobkom 1000.

Do poľa potrebujeme pristupovať: čítať obsah jednotlivých políčok a zapisovať na ne. Navyše však občas potrebujeme nejaký úsek poľa *reverznúť*, čiže obrátiť poradie prvkov v ňom. Na nasledujúcom obrázku je príklad takejto operácie: hore je nejaké pole a dole je nové pole, ktoré z neho dostaneme, keď reverzneme úsek začínajúci na políčku 4 a končiaci na políčku 7. (Pripomíname, že najľavejšie políčko má číslo 0.)

3	1	4	1	5	9	2	6	5	3	5	8	9	7	9	3
3	1	4	1	6	2	9	5	5	3	5	8	9	7	9	3

Všetky úseky, ktoré budeme reverzovať, budú mať začiatok na políčku s číslom deliteľným 1000 a taktiež ich dĺžka bude vždy násobkom 1000.

Súťažná úloha

Napište program, ktorý bude podľa zadaných inštrukcií meniť obsah nášho poľa a simulovať reverzy jeho častí. Poradíme vám ešte, že na získanie plného počtu bodov treba vymyslieť šikovný spôsob ako robiť reverzy. Riešenie, ktoré bude zakaždým všetky prvky jeden po druhom presúvať z miesta na miesto, bude na veľkých vstupoch potrebovať veľmi veľa času.

Formát vstupu a výstupu

Dostanete od nás 5 vstupných súborov, označených `1.txt` až `5.txt`. Každý z nich má nasledujúci formát:

V prvom riadku vstupu je číslo n : dĺžka poľa. Toto číslo je násobkom 1000.

Na začiatku každé políčko obsahuje svoje vlastné číslo. Teda ak sa naše pole volá napríklad A , tak na začiatku platí $A[0] = 0$, $A[1] = 1$, atď., až $A[n - 1] = n - 1$.

V druhom riadku vstupu je číslo q , udávajúce počet inštrukcií, ktoré treba v zadanom poradí vykonať.

Zvyšok vstupu tvorí q riadkov, z ktorých každý obsahuje jednu inštrukciu. Inštrukcie majú nasledovný tvar:

- „1 x y “: do poľa na políčko s indexom x ulož hodnotu y
- „2 p q “: reverzni úsek poľa, tvorený políčkami s číslami $1000p$ až $(1000q - 1)$.
- „3 a b “: do výstupného súboru vypíš riadok s hodnotami, ktoré sú práve v našom poli na indexoch a až b

Všetky inštrukcie sú korektné. Presnejšie, vždy platia nasledujúce obmedzenia:

- $0 \leq x < n$ (všetky čísla políčok sú z rozsahu od 0 po $n - 1$)
- $0 \leq y \leq 10^9$ (všetky hodnoty ukladané do poľa sa pohodlne zmestia do bežnej celočíselnej premennej)
- $0 \leq p < q \leq n/1000$ (dvojica p, q popisuje neprázdny úsek políčok ktorého hranice sú násobky 1000)
- $0 \leq a \leq b < n$ (vypisujeme vždy neprázdny úsek políčok)

Veľkosti vstupov

V jednotlivých vstupoch má pole dĺžku 1 000, 7 000, 300 000, 1 000 000 a 3 000 000. Počet inštrukcií je porovnateľný s dĺžkou poľa. Celkový počet vypísaných čísel (inštrukciou 3) je vždy rozumne malý, aby veľkosť správneho výstupu neprekročila 15 MB.

Odvzdávanie riešení

Toto je praktická úloha. Napište v ľubovoľnom programovacom jazyku program, ktorý ju rieši.

Zo stránky <http://oi.sk/> stiahnite ZIP archív obsahujúci 5 testovacích vstupov, nazvaných `1.txt` až `5.txt`.

Vyrobte k čo najviac vstupom správne výstupy a uložte ich do súborov `sol1.txt` až `sol5.txt`.

Odvzdajte ZIP archív obsahujúci zdrojový kód vášho programu a tieto výstupné súbory.

Za každý správny výstupný súbor získate 2 body.



Príklad

vstup

```
2000
10
3 0 7
1 2 424242
1 2 474747
3 0 7
2 0 1
3 2 5
2 0 2
3 2 5
1 999 12345
3 997 1002
```

výstup

```
0 1 2 3 4 5 6 7
0 1 474747 3 4 5 6 7
997 996 995 994
1997 1996 1995 1994
1002 1001 12345 0 1 474747
```

Vyššie uvedený príklad vstupu začína tým, že sa dozvieme, že $n = 2000$. Naše pole má teda dĺžku 2000. Na začiatku sú v ňom uložené hodnoty $0, 1, 2, \dots, 1999$ v tomto poradí.

V druhom riadku vstupu sa dozvieme, že treba postupne spracovať $q = 10$ inštrukcií. Tie vyzerajú nasledovne:

1. Inštrukcia „3 0 7“ nám hovorí „vypíš prvky na indexoch 0 až 7“. Vypísané hodnoty vidíme v prvom riadku vyššie uvedeného správneho výstupu.
2. Inštrukcia „1 2 424242“ nám hovorí „na pozíciu 2 zapíš hodnotu 424242“. Hodnota uložená na pozícii 2 sa zmení z 2 na 424242.
3. Inštrukcia „1 2 474747“ nám hovorí „na pozíciu 2 zapíš hodnotu 474747“. Hodnota uložená na pozícii 2 sa zmení z 424242 na 474747.
4. Inštrukcia „3 0 7“ nám opäť hovorí, že máme vypísať prvých 8 políček. Výsledok, už s novou hodnotou 474747, vidíme v druhom riadku správneho výstupu.
5. Inštrukcia „2 0 1“ nám hovorí, že máme reverznúť úsek tvorený políčkami s číslami 0 až 999.
6. Inštrukcia „3 2 5“ nám hovorí, že máme vypísať políčka s číslami 2, 3, 4, 5. Reverzom, ktorý sme vykonali v predchádzajúcom kroku, sa na tieto políčka dostali nové hodnoty, ktoré vidíme v treťom riadku správneho výstupu.
7. Inštrukcia „2 0 2“ nám hovorí, že máme reverznúť úsek tvorený políčkami s číslami 0 až 1999, čiže celé naše pole.
8. Inštrukcia „3 2 5“ po reverze vypíše nové hodnoty, ktoré sa presunuli na tieto políčka.
9. Inštrukcia „1 999 12345“ uloží novú hodnotu 12345 na políčko s číslom 999.
10. Inštrukcia „3 997 1002“ nám vypíše šesť hodnôt v strede poľa.



B-I-3 Hra s kartami

Baška s Vlejdodom trávili toto leto v švajčiarskom Zurichu, kde programovali pre Google. Všetko bolo super, až na ten jeden deň, keď im doma vypadol internet. Keďže vonku strašne pršalo a internet nefungoval, neostávalo im nič iné, ako sa začať hrať s kartami.

Najskôr hrali Prší, potom Žolíka a nakoniec aj Vojnu. Napriek tomu sa rýchlo začali nudiť. A keď postavili už jedenáste poschodie kartového domčeka, rozhodli sa vyskúšať niečo nové. Zobrali strašne veľa kariet z rôznych hier a zamiešali ich dokopy. Potom, obrázkami nahor, ich rozložili do jedného dlhého radu.

„Dve rovnaké karty vedľa seba!“ vykrikuje Baša a ukazuje doprostred radu.

„Aha,“ vraví Vlejd, „tu sú hneď tri rovnaké karty ležiace za sebou.“

„To je slabé. Pozri na týchto osem kariet, ktoré ležia vedľa seba, Všetky sú rôzne.“ chváli sa Baška.

„Iba osem? Tu ich je trinásť a tiež sú všetky navzájom rôzne.“ podpichne ju Vlejd.

To už je ale Baška nahnevaná, že ju Vlejd stále prekonáva, a chcela by mu ukázať ešte dlhšiu takú postupnosť. Najlepšie tú najdlhšiu, nech ju už potom Vlejd nemôže prekonať. Viete jej pomôcť?

Súťažná úloha

Dostanete popis radu kariet, ktoré majú Baška s Vlejdodom vyložené pred sebou. Pre jednoduchosť budeme karty popisovať číslami – rovnaké čísla označujú rovnakú kartu, rôzne čísla označujú rôzne karty.

Vašou úlohou je v tejto postupnosti čísel nájsť najdlhší úsek za sebou idúcich čísel, v ktorom sa žiadne číslo nenachádza viac ako raz. V prípade, že je takýchto úsekov viacero, stačí nájsť ľubovoľný z nich.

Formát vstupu a výstupu

V prvom riadku vstupu je číslo n označujúce počet vyložených kariet a číslo m udávajúce najväčšie použité číslo karty.

V druhom riadku je n celých čísel oddelených medzerami. Tieto čísla reprezentujú vyložené karty v poradí v akom boli položené na stôl. Všetky tieto čísla sú z rozsahu od 1 po m .

Na výstup vypíšte dve čísla. Prvé je dĺžka najdlhšieho úseku navzájom rôznych kariet a druhé je ľubovoľná pozícia (číslované od 1), na ktorej takýto úsek začína.

Hodnotenie

Za ľubovoľné správne riešenie získate aspoň 3 body.

Až 5 bodov môžete získať za riešenie, ktoré zvládne efektívne vyriešiť vstupy, kde $n \leq 2\,000$ a $m \leq 30$.

Až 7 bodov môžete získať za riešenie, ktoré zvládne efektívne vyriešiť vstupy, kde $n \leq 10\,000$ a $m \leq 1\,000\,000$.

Plných 10 bodov získa riešenie, ktoré dokáže efektívne riešiť vstupy s $n \leq 1\,000\,000$ a $m \leq 1\,000\,000$.

Odovzdávanie riešení

Toto je teoretická úloha. Spísané riešenie vo formáte PDF odovzdajte pomocou webového rozhrania.

Príklad

vstup

10 50
1 50 1 4 7 7 9 50 47 9

výstup

4 2

Vyššie uvedený výstup hovorí, že najdlhší úsek čísel, ktoré sú všetky navzájom rôzne, má dĺžku 4 a začína na pozícii 2. Ide teda o úsek „50, 1, 4, 7“.

Správna by bola aj odpoveď 4 6, ktorá zodpovedá úseku „7, 9, 50, 47“.



B-I-4 Tabuľa

Mário prišiel do triedy, zotrel tabuľu a napísal na ňu čísla od 1 do 50, každé práve raz.

Kaja sa ide hrať s týmito číslami. Presne 49-krát zopakuje nasledujúci postup: vyberie si nejaké dve čísla na tabuli, obe ich zmaže, a následne na tabuľu zapíše hodnotu, o ktorú sa líšili. Ak teda zmazala čísla x a y , naspäť napíše číslo $|x - y|$, čiže absolútnu hodnotu ich rozdielu.

Počas hry sa môže stať, že bude na tabuli viackrát napísané to isté číslo. Ak napríklad Kaja hneď na začiatku zmaže čísla 6 a 9, napíše na tabuľu číslo 3. V tej chvíli teda budú na tabuli dve trojky. Keby potom zmazala obe trojky, napísala by namiesto nich na tabuľu číslo 0.

Súťažná úloha

Súťažnú úlohu sme rozdelili na štyri podúlohy. Hodnotené sú každá zvlášť, ale aj tak vám ich odporúčame riešiť v zadanom poradí.

- **Podúloha A (2 body).** Poradte Kaji, ako má hrať, aby jej na konci na tabuli ostalo číslo 1. (Existuje veľa riešení, stačí nájsť ľubovoľné jedno z nich.)
- **Podúloha B (2 body).** Aké najväčšie číslo môže Kaja (kedykoľvek počas hry, nie nutne na jej konci) zapísať na tabuľu? Prečo nikdy nezapíše žiadne väčšie číslo?
- **Podúloha C (3 body).** Vie Kaja hrať hru tak, aby jej na konci ostalo na tabuli číslo 14? Ak áno, ako? Ak nie, prečo?
- **Podúloha D (3 body).** Nájdite **úplne všetky** možné hodnoty, ktoré môžu byť na konci hry na tabuli. Samozrejme, svoje tvrdenie zdôvodnite. (Naozaj sa dá vyrobiť každá z vami uvedených hodnôt? Ako? A naozaj sa nedá vyrobiť už žiadna iná? Prečo?)

Odvzdávanie riešení

Toto je teoretická úloha. Spísané riešenie vo formáte PDF odovzdajte pomocou webového rozhrania.