Návodné úlohy k domácemu kolu OI: kategória A

Toto sú návodné úlohy k domácemu kolu 32. ročníka Olympiády v informatike. Ide teda o sadu ľahších úloh, ktoré tematicky súvisia so súťažnými úlohami. Riešenie týchto úloh môže byť dobrou prípravou na riešenie súťažných úloh. Za riešenia týchto návodných úloh nie sú žiadne body do súťaže.

A-I-1 Bežecké preteky

1. Máme dve polia reálnych čísel. V prvom sú veky c chlapcov, usporiadané od najmenšieho po najväčší. V druhom sú tak isto usporiadané veky d dievčat.

   Chceme vybrať dvojicu (chlapec,dievča) tak, aby bol chlapec starší ako dievča. Koľkými spôsobmi to vieme spraviť? Napíšte program, ktorý to spočíta.

   Existuje riešenie s časovou zložitosťou $O(c + d)$.

2. Máme pole A, v ktorom je uložená nejaká permutácia čísel 1 až n. Chceli by sme toto pole usporiadať. Smieme ho však meniť len jediným spôsobom: tak, že nájdeme nejaké i, pre ktoré momentálne platí $A[i] > A[i + 1]$, a takúto dvojicu prvkov vymeníme.

   Napr. pole $(3, 1, 5, 2, 4)$ vieme zmeniť na pole $(3, 1, 2, 5, 4)$ a to následne zmeniť na pole $(1, 3, 2, 5, 4)$.

   Dokážte, že opakovaním vyššie popísanej zmeny vieme usporiadať úplne ľubovoľné pole.

3. Ďalej dokážte, že v predchádzajúcej úlohe vlastne príliš nezáleží na poradí, v akom tie výmeny robíme – bez ohľadu na to, čo kedy vymieňame, celkový počet výmen potrebný na usporiadanie konkrétneho poľa je vždy rovnaký.

4. Napíšte program, ktorý načíta konkrétne pole A (obsahujúce permutáciu čísel 1 až n) a spočíta, koľko výmen po sebe idúcich prvkov treba na jeho usporiadanie.

   (Existuje ľahké riešenie s časovou zložitosťou kvadratickou od n, existujú však aj výrazne efektívnejšie riešenia.)

5. Dané sú dve rovnako dlhé polia A a B. Každé z nich obsahuje nejakú permutáciu čísel 1 až n.

   Chceme pole A prerobiť na pole B. V každom kroku však smieme len vymeniť nejaké dva susedné prvky poľa A. Napíšte program, ktorý vypočíta, koľko najmenej krokov stačí na prerobenie A na B.

A-I-2 Rekonštrukcia školy

1. Vyriešte jednoduchšiu verziu súťažnej úlohy: učiteľ má rozmery 1×1 a nikdy nesmie stúpiť na prekážku.

2. Čo ak má učiteľ rozmery 1×1 a nanajvýš raz smie stúpiť na prekážku?

3. A čo ak má stále učiteľ rozmery 1×1, ale po ceste už smie stúpiť na nanajvýš 7 rôznych prekážok?

4. V zadaní súťažnej úlohy sa nachádza nasledujúca veta: „Na to jediné políčko s prekážkou smie učiteľ počas svojej cesty stúpiť ľubovoľne veľakrát, dokonca z neho môže odísť a neskôr sa naň zase vrátiť, ak chce.“

   Naozaj sa to niekedy oplatí? Nájdite konkrétny vstup, pre ktorý sa učiteľ vie dostať z kancelárie do triedy, musí však cestou stúpiť na prekážku, potom z nej odísť, a niekedy neskôr znova stúpiť na *tú istú* prekážku.

A-I-3 Moderné umenie

1. Daných je n bodov v rovine a uhol φ. Napíšte program, ktorý zistí, ktorý spomedzi daných bodov bude mať najväčšiu x-ovú súradnicu, ak celú rovinu otočíme okolo bodu $(0, 0)$ o uhol φ v smere hodinových ručičiek.

2. Daných je n bodov v rovine. Napíšte program, ktorý spomedzi nich vyberie nejakú dvojicu A, B takú, že všetky ostatné body ležia na tej istej strane priamky AB (prípadne rovno na nej).

3. Vyriešte ľahšiu verziu súťažnej úlohy: Netreba nájsť konkrétny uhol, stačí len zistiť, či nejaký taký uhol existuje.

4. Daných je n bodov v rovine. Predstavme si, že celú rovinu otáčame okolo bodu $(0, 0)$, až kým ju neotočíme o $360°$. Počas otáčania neustále sledujeme, ktorý bod má momentálne najväčšiu x-ovú súradnicu.

   Napíšte program, ktorý nájde všetky body, ktoré niekedy počas otáčania roviny budú mať najväčšiu x-ovú súradnicu.

A-I-4 Stromochod

1. Strom je *cesta idúca doľava* ak platí, že každý vrchol má len ľavého syna, až na jednu výnimku – jeden z vrcholov stromu je list a ten už nemá vôbec žiadneho syna.

   Napíšte program pre stromochod, ktorý v koreni nastaví značku `ok` vtedy, ak je strom cestou idúcou doľava.

2. Strom je *cesta idúca cikcak* ak platí, že koreň má len ľavého syna, vrchol pod koreňom má len pravého syna, ten má len ľavého syna, a tak ďalej, až kým neprídeme do jediného listu.

   Napíšte program pre stromochod, ktorý v koreni nastaví značku `ok` vtedy, ak je strom cestou idúcou cikcak.

3. Máme všeobecný binárny strom. Napíšte program pre stromochod, ktorý nastaví značku `parny` práve v tých vrcholoch, ktoré majú párnu vzdialenosť od koreňa (vrátane koreňa samotného, keďže aj nula je párna).

4. Napíšte program pre stromochod, ktorý v koreni nastaví značku `ok` vtedy, ak strom obsahuje aspoň jeden vrchol, ktorý má aj ľavého, aj pravého syna.

5. Napíšte program pre stromochod, ktorý v koreni nastaví značku `ok` vtedy, ak strom obsahuje nanajvýš 47 vrcholov, z ktorých každý má aj ľavého, aj pravého syna.

6. Strom má *tvar písmena V*, ak platia nasledovné podmienky:
   – koreň má aj ľavého, aj pravého syna
   – ľavý syn koreňa má len ľavého syna, ten má len ľavého syna, a tak ďalej až po list
   – symetricky, pravý syn koreňa má len pravého syna, ten má len pravého syna, a tak ďalej až po list
   – oba listy sú rovnako ďaleko od koreňa

   Najmenší strom tvaru písmena V má tri vrcholy: koreň a pod ním dva listy. Druhý najmenší takýto strom má 5 a tretí najmenší takýto strom má 7 vrcholov.

   Napíšte program pre stromochod, ktorý v koreni nastaví značku `ok` vtedy, ak ide o strom tvaru písmena V.

7. V príklade v študijnom texte sme si ukázali, ako vie stromochod celý strom *prehľadať do hĺbky*.

   Predstavme si, že by sme všetky vrcholy stromu očíslovali od 1 po nejaké n (kde n je ich počet) v poradí, v akom boli počas toho prehľadávania prvýkrát navštívené. Číslo 1 teda dostane koreň stromu, keďže tam začíname. Číslo vrcholu v označíme $c(v)$.

   Rozmyslite si, že samotný stromochod si tieto čísla **nevie** počítať a zapisovať, pretože má k dispozícii (aj v svojej pamäti, aj v každom vrchole stromu) len konštantný počet bitov.

   Upravte program zo študijného textu tak, aby si v každom vrchole v zapamätal v premennej `cislo` hodnotu ($c(v)$ mod 4), teda zvyšok, ktorý dáva jeho číslo po delení štyrmi. (To si už dovoliť môžeme, v každom vrchole nám na to stačia dva bity.)

Návodné úlohy k domácemu kolu OI: kategória B

Toto sú návodné úlohy k domácemu kolu 32. ročníka Olympiády v informatike. Ide teda o sadu ľahších úloh, ktoré tematicky súvisia so súťažnými úlohami. Riešenie týchto úloh môže byť dobrou prípravou na riešenie súťažných úloh. Za riešenia týchto návodných úloh nie sú žiadne body do súťaže.

B-I-1 Pokémoni

1. Oplatí sa niekedy opustiť pokéstop a mať pri sebe vtedy menej ako m pokélôpt? Prečo?

2. Peťo práve prichádza na pokéstop a nesie si so sebou nejakých pokémonov a jednu prázdnu pokéloptu. Stretne ho Jarka a pýta sa ho: „Tak čo, pochytal si všetkých pokémonov čo si stretol?“ Peťo na to: „Nie, práve som prešiel okolo jedného Pidgeyho, ktorého som nechal ujsť.“

   Mohol v ten deň Peťo nachytať pokémonov optimálnym spôsobom? Prečo?

3. Máme reťazec, v ktorom sú nejaké písmená (od `a` po `z`) a nejaké medzery. Súvislé úseky písmen voláme slová.

   Napíšte program, ktorý načíta takýto reťazec do premennej a následne ho rozdelí na jednotlivé slová a vypíše každé slovo na nový riadok.

B-I-2 Obracanie poľa

1. Máme pole veľkosti n. Do poľa potrebujeme pristupovať: čítať obsah jednotlivých políčok a zapisovať na ne. Navyše však občas potrebujeme **úplne celé pole** *reverznúť*, čiže obrátiť poradie prvkov v ňom.

   Napíšte program, ktorý každú takúto operáciu (čítanie políčka, zápis na políčko, aj reverz celého poľa) zvládne spraviť v konštantnom čase.

2. V rade je uložených a škatúľ. Pozície, na ktorých škatule stoja, majú čísla od 0 po $a-1$.

   V každej škatuli je b priehradiek. Priehradky v každej škatuli sú očíslované od 0 po $b-1$. V každej priehradke je uložené nejaké písmeno (od `a` po `z`), pričom na začiatku je úplne všade uložené písmeno `a`.

   (Zaujímavý je pre nás napríklad prípad kedy $a = 1000$ aj $b = 1000$.)

   Ku jednotlivým písmenkám chceme vedieť pristupovať: čítať ich a meniť ich. (Napr.: „v škatuli na pozícii 47 ulož do priehradky 42 písmeno `x`“.)

   Okrem toho chceme vedieť vymieňať dvojice škatúľ. (Napr.: „vymeň škatule na pozíciách 74 a 24“.)

   Napíšte program, ktorý každú takúto operáciu (čítanie, zápis aj výmenu dvoch škatúľ) zvládne spraviť v konštantnom čase.

B-I-3 Hra s kartami

1. Olívia má prázdne vrece. Peťo má guličky. Na každej guličke je číslo od 1 do 10^6. Guličiek s každým číslom má Peťo obrovskú zásobu.

   Peťo občas hodí Olívii do vreca nejakú guličku, a občas sa Olívie opýta: „Koľko rôznych čísel sa nachádza na guličkách vo tvojom vreci?“

   Táto hra Olíviu nudí. Napíšte jej program, ktorý bude túto hru hrať za ňu. Program teda má spracúvať postupnosť inštrukcií, pričom každá inštrukcia je buď „pridaj guličku s číslom x“ alebo „oznám počet *rôznych* guličiek vo vreci“.

   Nájdite riešenie, ktoré si na začiatku inicializuje pamäť a následne každú operáciu spracuje v malom konštantnom čase.

2. Upravte program z predchádzajúcej úlohy tak, aby mohol Peťo aj vyberať guličky z vreca. Pribudne teda nová inštrukcia „vyber guličku s číslom x“, pričom takúto inštrukciu budeme volať len ak sa aspoň jedna taká gulička vo vreci skutočne nachádza.

Príklad:

```
>>> roznych
0
>>> vloz 4
>>> vloz 4
>>> roznych
1
>>> vloz 7
>>> vloz 47
>>> roznych
3
>>> vyber 4
>>> roznych
3
>>> vyber 4
>>> roznych
2
```

B-I-4 Tabuľa

1. Vyskúšajte si ručne (skúšaním všetkých možností), čo by sa dialo, keby v súťažnej úlohe bolo čísel na tabuli na začiatku menej. Čo keby to neboli čísla od 1 po 50, ale len po 2, 3, 4, či 5? Aké výsledky sa v ktorom z týchto prípadov dajú dosiahnuť?

2. Na tabuli sú čísla od 1 po 50. V každom kroku môže Kaja nejaké dve čísla zmazať a nahradiť ich poslednou cifrou ich súčtu. (Ak napríklad zmaže 3 a 4, napíše 7. Ak zmaže 8 a 8, napíše namiesto nich 6.)

   Celý proces skončí, keď už je na tabuli len jediná cifra. Aká cifra to môže byť? Nájdite všetky možnosti!

3. Máme nasledovný výraz: $\pm 1 \pm 2 \pm 3 \pm \cdots \pm 99 \pm 100$

   Každé $\pm$ treba zmeniť buď na plus alebo na mínus.

   a) Nájdite nejaký spôsob, ako to spraviť tak, aby vyšiel výsledok 0.

   b) Marienka nejak zmenila každé $\pm$ na $+$ alebo $-$, čím dostala výraz s nejakou nám neznámou hodnotou. Vladko potom zmenil jedno znamienko (z $+$ na $-$ alebo naopak). O koľko sa mohla zmeniť hodnota výrazu?

   c) Vieme v pôvodnom výraze zmeniť všetky $\pm$ na $+$ a $-$ tak, aby vyšiel výsledok 47?