

Domáca úloha č. 3

1-AIN-105, Zima 2025

Termín: 24.11.2025, 22:00

Skôr ako sa pustíte do riešenia, prečítajte si všeobecné pokyny na konci zadania.

1. [20 bodov] **Sú algoritmy správne?** Zistite, či pre uvedený problém je daný greedy algoritmus správny. Dokážte správnosť algoritmu alebo uveďte kontrapríklad.

- a) **Podivná hra.** Na stole je v rade vyložených $2n$ kartičiek s číslami. Hru hrajú dvaja hráči, pričom každý z nich si môže v každom ťahu zobrať jednu z dvoch krajných kartičiek. Vyhráva ten, kto má na konci väčší súčet čísel na svojich kartičkách. Cieľom prvého hráča je vyhrať, nech druhý hráč hrá akokoľvek.

Algoritmus pre prvého hráča. Vždy keď je na ťahu, zoberie z dvoch krajných kartičiek tú s väčším číslom. Poznámka: Ak chcete ukázať, že algoritmus je správny, musí prvý hráč pomocou neho vyhrať, nech druhý hráč hrá akokoľvek. Ak chcete ukázať, že algoritmus nie je správny, stačí nájsť príklad hry, kedy druhý hráč porazí prvého hráča.

- b) **Plátanie hadíc.** Na nemenovanej fakulte požiarňa inšpekcia našla deravú požiarňu hadicu. Keďže peňazí je málo, údržba sa rozhodla hadicu zaplatať. Hadica má k malých dier vo vzdialenostiach $d_1 < d_2 < \dots < d_k$ centimetrov od začiatku. Máme k dispozícií záplaty, ktoré pokrývajú 5 cm dĺžky. Cieľom je zalepiť všetky diery s použitím najmenšieho možného počtu záplat.

Algoritmus. Položíme ľavý koniec záplaty na najľavejšiu diery. Zmažeme zo zoznamu všetky diery pokryté záplatou a opakujeme, až kým neostanú žiadne diery.

2. [20 bodov] **Detské tábory.** Pre účely tohto príkladu očísľujeme dni obdobia prázdnin celými číslami od 1 po m . Na fakulte sa počas prázdnin bude konať n táborov, pričom i -ty tábor sa bude konať odo dňa s_i po deň f_i vrátane ($s_i \leq f_i$). Úrad pre ochranu práv malých detí chce skontrolovať dodržiavanie práv detí vo všetkých táboroch. Každý deň, keď kontrolór navštívi fakultu, skontroluje všetky tábory, ktoré práve prebiehajú. Aký je najmenší možný počet návštev, ktoré kontrolór musí vykonať, aby skontroloval všetky tábory?

Príklad: Uvažujme $n = 5$ táborov, pričom $s_1 = 1, f_1 = 6, s_2 = 1, f_2 = 9, s_3 = 10, f_3 = 11, s_4 = 7, f_4 = 9, s_5 = 3, f_5 = 7$. Kontrolór musí navštíviť fakultu trikrát, jedno z možných riešení je návšteva v dňoch 4, 7 a 10.

- a) Uvažujte nasledujúci greedy algoritmus:

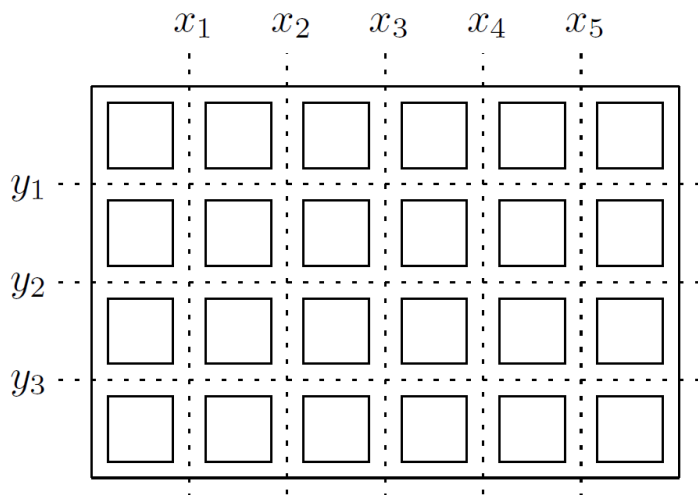
- (a) Nájdite deň x , ktorý sa prekrýva s najväčším možným počtom táborov. Ak existuje viacero takých dní, zoberite z nich ten s najmenším číslom.
- (b) Vyberte x ako súčasť riešenia.
- (c) Vymažte všetky tábory, ktoré zahŕňajú deň x .
- (d) Opakujte, až kým nie sú skontrolované všetky tábory.

Ukážte, že tento algoritmus **nie je správny**, t.j. nájdite kontrapríklad, na ktorom algoritmus nenájde najmenší možný počet kontrolných dní.

- b) Navrhните iný greedy algoritmus, ktorý nájde optimálne rozmiestnenie kontrolných dní. Dokážte, že tento algoritmus **je správny**, t.j. že počet kontrolných dní bude najmenší možný.

3. [20 bodov] **Lámanie čokolády (programátorská úloha).** Máme čokoládu, ktorú by sme chceli rozlámať na jednotlivé tabličky. Problém je, že táto čokoláda je tvrdá ako kameň a teda sa zle láme. Pozdĺž niektorých hrán sa čokoláda láme oveľa horšie ako pozdĺž iných. Chceli by sme ju rozlámať tak, aby sme na to vynaložili čo najmenej námahy.

Sú dané rozmery čokolády r a s (koľko tabličiek je čokoláda vysoká a široká). Tiež máme kladné čísla $x_1, x_2, \dots, x_{s-1}$ a $y_1, y_2, \dots, y_{s-1}$, ktoré nám hovoria, koľko námahy musíme vynaložiť na prelomenie hrany. Presnejšie, pri lamaní pozdĺž i -tej zvislej čiary treba vynaložiť x_i námahy, a podobne pre vodorovné čiary.



Čokoládu lámeme postupne, krok za krokom. V každom kroku si vyberieme jeden jej kúsok, zvolíme si na ňom čiaru, podľa ktorej ho chceme prelomiť, vynaložíme námahu zodpovedajúcu dotyčnej čiare, a tým zlomíme aktuálny kúsok na dva nové.

Všimnite si, že pre niektoré čiary budeme námahu danú na vstupe vynakladať viackrát. Napríklad ak polámeme čokoládu najprv podľa vodorovných čiar a potom podľa zvislých čiar, naša námaha bude $y_1 + y_2 + \dots + y_{r-1} + r(x_1 + x_2 + \dots + x_{s-1})$, pretože každý z r vodorovných pásikov potrebujeme prelomiť $(s - 1)$ -krát (viď obrázok). Vašou úlohou je napísať program, ktorý vyráta, akú najmenšiu celkovú námahu treba vynaložiť na rozlamanie čokolády.

Vstup. Na prvom riadku vstupu sa nachádzajú dve medzerou oddelené čísla s, r . Platí, že $1 \leq r, s \leq 100\,000$. Na druhom riadku sa nachádzajú medzerou oddelené čísla x_i . Na treťom riadku sa nachádzajú medzerou oddelené čísla y_i .

Výstup. Vypíšte jediné číslo - námahu, ktorú potrebujeme vynaložiť na rozlamanie čokolády. Keďže toto číslo môže byť veľmi veľké, vypíšte iba jeho zvyšok po delení 1 000 000 007.

Príklad vstupu:

```
1 4
4 1 2
```

Príklad výstupu:

7

Túto čokoládu môžeme prelámať v ľubovoľnom poradí

Príklad vstupu:

```
2 3
1
1 2
```

Príklad výstupu:

6

Ak najprv prelomíme čokoládu pozdĺž čiar s námahou 2, ostane nám 2 kúsky. Jeden z nich rozlámeme s námahou 1 a druhý s námahou 3.

Príklad vstupu:

```
6 4
2 1 3 1 4
4 1 2
```

Príklad výstupu:

42

4. [20 bodov] **Preskáč cez chodník (programátorská úloha).** Rekonštrukcia našej fakulty sa dotkla aj chodníkov pred fakultou. Rozhodlo sa, že na tieto chodníky musí ísť nová dlažba. Problém je, že medzi tým, čo robotníci vybrali starú dlažbu a položili novú začalo pršať a chodník sa zmenil na veľkú mláku vody. Našťastie (pre nás) ostalo z vody trčať niekoľko zabudnutých kúskov dlaždíc.

Dlaždice sú všetky v jednom rade, aj keď nie nutne v pravidelných rozstupoch. Dlaždíc je $n - 1$. Očíslujeme si ich za radom smerom od nás od 1 po $n - 1$. Navyše náš kraj chodníka nazveme **dlaždica 0** a opačný kraj nazveme **dlaždica n**. (Pri riešení úlohy predpokladajte, že každá dlaždica je bod a všetky tieto body ležia na jednej priamke.)

Vzdialenosti medzi jednotlivými dlaždicami si môžeme popísať nasledovne: existuje nejaké číslo m také, že pre každé i je vzdialenosť medzi dlaždicami i a $i + 1$ rovná $1 + (i^2 \bmod m)$.

Napríklad teda ak $m = 10$, tak je vzdialenosť medzi dlaždicami 6 a 7 rovná $1 + (36 \bmod 10) = 7$.

Tvojou úlohou je dostať sa na druhú stranu tak, že spravíš nanajvýš j skokov. Obávaš sa však, že na to nevieš robiť dostatočne dlhé skoky.

Vypočítaj najmenšie x také, že osoba, ktorá vie robiť skoky ľubovoľnej dĺžky nanajvýš rovnej x , vie po dlaždiciach preskákať cez túto mláku tak, že spraví nanajvýš j skokov.

(Začína sa na dlaždici 0, každý skok musí pristáť na niektorej dlaždici, posledný skok musí pristáť na dlaždici n .)

Vstup. V jedinom riadku vstupu sú čísla n a m popisujúce chodník a číslo j udávajúce maximálny počet skokov. Platí $1 \leq j \leq n \leq 100\,000$ a $1 \leq m \leq 10^9 + 7$.

Výstup. Vypíšte jeden riadok a v ňom jedno číslo: najmenšiu dostatočnú maximálnu dĺžku skoku. (Dá sa ukázať, že toto číslo je vždy celé.)

Príklad vstupu:

10 1 1

Príklad výstupu:

10

Rátajúc aj oba kraje, máme 11 dlaždíc. Medzi každými dvoma po sebe idúcimi je vzdialenosť 1, čiže celková šírka chodníka je 10. Ak ju chceme prejsť jedným skokom, musíme teda vedieť skákať aspoň do vzdialenosti 10.

Príklad vstupu:

10 1 3

Príklad výstupu:

4

Ten istý chodník, chceme ho prejsť nanajvýš tromi skokmi. Človek, ktorý vie skákať nanajvýš do vzdialenosti 3, sa tromi skokmi dostane nanajvýš na dlaždicu 9, chodník teda neprejde. Vedieť skákať do vzdialenosti 4 už stačí.

Príklad vstupu:

10 1234567 10

Príklad výstupu:

82

Pre tento chodník sú vzdialenosti medzi dlaždicami postupne rovné 1, 2, 5, 10, 17, 26, 37, 50, 65 a 82. Chceme ho preskákať nanajvýš 10 skokmi.

Zjavne potrebujeme vedieť skákať aspoň do vzdialenosti 82, aby sme sa dostali z dlaždice 9 na druhý kraj. No a takáto dĺžka skoku je aj dostatočná, keďže napríklad môžeme postupne skočiť na každú dlaždicu.

(Pre maximálnu dĺžku skoku 82 sa už zjavne vieme dostať cez tento chodník aj na menej ako 10 skokov. Rozmyslite si, že toto nijak neovplyvňuje skutočnosť, že 82 je správnou odpoveďou na našu otázku.)

Príklad vstupu:

10 1234567 4

Príklad výstupu:

87

Cez chodník z minulého príkladu sa chceme dostať na nanajvýš štyri skoky. Najmenšia maximálna dĺžka skoku, pre ktorú to ide, je 87. Vtedy to vieme spraviť napríklad tak, že skočíme z dlaždice 0 na dlaždicu 6, odtiaľ na dlaždicu 8, potom na dlaždicu 9 a na záver na dlaždicu 10. Prvý skok má dĺžku $1 + 2 + 5 + 10 + 17 + 26 = 61$, druhý $37 + 50 = 87$, tretí 65 a štvrtý 82.

Príklad vstupu:

12345 1000000007 2

Príklad výstupu:

313555165872

Všeobecné pokyny

Čítajte poriadne všetko, čo je napísané čierne na bielom. Riešenia, ktoré odovzdáte, musia byť vaše vlastné. Neopisujte a nesnažte sa nájsť riešenia v literatúre alebo na internete! Ak ste v priebehu riešenia úlohy diskutovali so spolužiakmi alebo s ChatGPT, napíšte vlastné riešenie s časovým odstupom a v riešení uveďte s kým/čím ste diskutovali (bližšie informácie na stránke predmetu).

60 bodov je považovaných za 100%. Pokiaľ z úlohy získate viac ako 60 bodov, tieto body navyše sú bonusové a zarátavajú sa vám do celkovej známky. **Riešenie písomnej úlohy odovzdané v anglickom jazyku aspoň za polovicu bodov získa ďalšie 2 bonusové body navyše.**

Písomné úlohy. Písomné úlohy odovzdávajú na adrese <https://vektor.fmph.uniba.sk> ako PDF súbory v stanovenom termíne. **Každý príklad odovzdajte v osobitnom PDF súbore. Preferujeme riešenia napísané na počítači a vyexportované do PDF súboru**, ručne písané skenované riešenia sú často ťažko čitateľné. Na neskoro odovzdané riešenia sa nebude prihliadať.

Píšte riešenia takým spôsobom, aby obsahovali všetku potrebnú informáciu na pochopenie vášho riešenia, ale súčasne aby boli stručné a ľahko pochopiteľné. Všetky tvrdenia je potrebné zdôvodniť (a to aj v prípade, že to nie je explicitne napísané v zadaní).

Ak sa v zadaní požaduje vyriešenie algoritmickej úlohy, odovzdajte najlepší algoritmus, aký viete navrhnúť. Základným kritériom na hodnotenie bude *správnosť algoritmu*, druhým kritériom bude jeho *časová, prípadne pamäťová zložitosť*. Správny ale pomalý algoritmus dostane podstatne viac bodov ako algoritmus, ktorý je síce rýchly, ale nedá správnu odpoveď na každý vstup. Neefektívne algoritmy spĺňajúce podmienky zadania dostanú cca 50% bodov. Súčasťou vášho riešenia musia byť nasledujúce časti:

- Najprv popíšte hlavnú myšlienku algoritmu.
- Vyjadrite algoritmus formou pseudokódu.
- Ak to nie je zrejmé na prvý pohľad, ukážte že váš algoritmus je správny.
- Nezabudnite na analýzu zložitosti algoritmu.

Ak nie je povedané inak, logaritmy majú základ 2.

Programátorské úlohy. Pri programátorských úlohách je úlohou odovzdať len funkčný program, nie je vyžadované písomné riešenie. Riešenie odovzdávate priamo na adrese <https://vektor.fmph.uniba.sk>, kde bude okamžite otestované na niekoľkých sadách vstupov a dozviete sa, koľko bodov získalo. Body získate, keď všetky vstupy z danej sady vyriešite správne v časovom limite. Riešenie môžete odovzdávať aj viackrát, hodnotí sa najlepšie riešenie odovzdané v stanovenom termíne, počet pokusov nie je obmedzený.

Domácu úlohu môžete odovzdať v ktoromkoľvek podporovanom programovacom jazyku. Upozorňujeme však, že vzorové riešenia sú ladené pre C++ a Python. Pri použití iných jazykov negarantujeme, že vaše riešenie dosiahne plný počet bodov. Podrobnosti o tom, ako má váš program vyzerieť (vrátane zoznamu podporovaných programovacích jazykov), nájdete na stránke <https://judge.ksp.sk/docs/co-odovzdavat/>.