

# Domáca úloha č. 3

1-AIN-105, Zima 2020

Termín: 19.11.2018, 22:00, google classroom

Skôr ako sa pustíte do riešenia domácej úlohy, oboznámte sa so všeobecnými pokynmi, ktoré sú priložené na konci tohto dokumentu. Riešenia, ktoré odovzdáte, musia byť vaše vlastné. Neopisujte a nesnažte sa nájsť riešenia v literatúre alebo na internete!

1. [20 bodov] Pre každú z nižšie uvedených rekurencií odvodte čo najtesnejší asymptotický horný odhad pre  $T(n)$ . Predpokladajte, že  $T(1) = \Theta(1)$  and  $T(0) = \Theta(1)$ . V prípade, že sa rozhodnete aplikovať master theorem, môžete zanedbať dolné a horné celé časti. Zdôvodnite svoje odpovede.

- a)  $T(n) = 7T(\lfloor n/2 \rfloor) + n^2$
- b)  $T(n) = 16T(\lfloor n/4 \rfloor) + n^2$
- c)  $T(n) = 4T(\lfloor n/2 \rfloor) + n^2\sqrt{n}$
- d)  $T(n) = T(\lfloor \sqrt{n} \rfloor) + 1$

2. [20 bodov] **Stavba diaľnice.** Projekt stavby diaľnice je rozdelený na  $n$  úsekov, pričom  $i$ -ty úsek bude trvať postaviť  $t_i$  času. Máte k dispozícii  $k$  partií stavbárov. Ak partia začne stavať úsek  $j$ , postaví ho za čas  $t_j$ , potom sa presunie na stavbu úseku  $j + 1$ , ktorý postaví za čas  $t_{j+1}$ , atď., až kým nenarazí na postavený úsek alebo koniec diaľnice; v takom prípade partia odíde domov a už sa na stavbu nevráti. Diaľnica je dostavaná, keď posledná partia odíde domov.

Navrhnite algoritmus, ktorý určí, na ktorom úseku má ktorá partia začať, aby sme celú diaľnicu postavili v najkratšom možnom čase (najlepšie už do volieb).

**Hint:** Ako by ste rozmiestnili partie, keby ste vedeli koľko bude stavba diaľnice trvať?

3. [20 bodov] **Programátorská úloha** (viď všeobecné pokyny). Šikovný predavač kníh si všimol, že keď kupujúcemu ponúkne knihu a správne odhadne cenu, tak ju kupujúci kúpi. V stánku na vianočných trhoch má na kope knihy a podľa oblečenia kupujúceho vie odhadnúť, aký je kupujúci prachatý. Ak je kupujúceho prachatosť  $p$  a hodnota knihy je  $h$ , treba ju kupujúcemu ponúknuť za cenu  $ph$  bubákov a kupujúci ju kúpi.

Najprachatejším kupujúcim chceme tým pádom ponúkať tie najhodnotnejšie knihy. Na trhoch však nie je čas prehrávať sa všetkými knihami na kope a hľadať tú správnej hodnoty, preto predavač ohodnotí pohľadom kupujúceho a vyberie pre neho **jednu z troch najvrchnejších kníh** na kope.

V stánku máme  $n$  kníh o hodnotách  $h_1, h_2, \dots, h_n$  (hodnoty kníh sú usporiadané od vrchu kopy až po spodok) a v rade stojí  $n$  kupujúcich, pričom ich prachatosť je  $p_1, p_2, \dots, p_n$  (v tomto poradí). Koľko najviac peňazí môže predavač utržiť za svoju kopy kníh?

**Formát vstupu:** V prvom riadku je číslo  $n$ . V druhom riadku sú medzerou oddelené hodnoty kníh  $h_1, h_2, \dots, h_n$ . V treťom riadku sú medzerou oddelené prachatosťi zákazníkov  $p_1, p_2, \dots, p_n$ .

**Formát výstupu:** Na výstupe vypíšete jediné číslo—najväčší možný obnos peňazí, ktorý môžete za knihy utržiť.

**Obmedzenia:** Môžete predpokladať  $n \leq 500$  a  $1 \leq h_i \leq 1000$  a  $1 \leq p_i \leq 1000$ .

**Príklad vstupu:**

4  
16 6 2 10  
3 8 12 9

**Príklad výstupu:**

336

Prvému kupujúcemu ponúkžeme tretiu knihu z vrchu o hodnote 2, za ktorú zaplatí 6 bubákov. Zostávajú nám knihy s hodnotami 16 6 10 (v tomto poradí). Druhému kupujúcemu ponúkžeme druhú knihu z vrchu, za ktorú zaplatí 48

bubákov a zostanú nám knihy s hodnotami 16 a 10. Tretí kupujúci zaplatí za vrchnú knihu 192 bubákov a posledný kupujúci zaplatí za vrchnú knihu 90 bubákov.

**Hint:** Porozmýšľajte nad dynamických programovaním, kde by podproblém reprezentoval stav kopy kníh.

**Hint 2:** Python je síce fajn jazyk, ale niekedy je lepšie použiť niečo rýchlejšie.

## Všeobecné pokyny

**Písomné úlohy.** 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.

Písomné úlohy odovzdávajú do Google Classroom pokiaľ možno *ako jeden PDF dokument*. Na neskoré odovzdané riešenia sa nebude prihliadať.

**Programátorské úlohy.** Pri programátorských úlohách je vašou úlohou odovzdať len funkčný program, nie je vyžadované písomné riešenie. Riešenie odovzdávate cez web stránku predmetu na <http://compbio.fmph.uniba.sk/vyuka/eaz/assignments.php>, kde bude okamžite otestované na niekoľkých vstupoch 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 posledné riešenie odovzdané v stanovenom termíne. Navyše si dajte pozor, či v systéme máte správne vyplnené meno a priezvisko. Zoznam programovacích jazykov a inštrukcie k tomu, ako presne má program vyzerieť, nájdete na stránke <https://testovac.ksp.sk/wiki/odovzdavanie-info/>.