



Záverečný test praktická časť



Ústav informatiky
Prírodovedecká fakulta
UPJŠ v Košiciach

Doplňujúce zdrojové kódy sú na stránke predmetu PAZ1b. Funkčnosť každého riešenia musí byť preukázaná spustením na testovacom vstupe - nespustiteľné riešenia neumožňujú zisk príslušných bodov. Body nebude možné udeliť pokiaľ študent nevie vysvetliť svoje riešenie.

Summer time

Začína skúškové obdobie a po ňom príde leto. Počas leta je vonku teplo, každý chce byť s kamarátmi, na brigáde alebo sa venovať svojej rodine a svojim hobby. Iný sa chcú opaľovať a užívať si slnko. Nezáleží, či máš peniaze a pôjdeš k moru alebo sa budeš opaľovať na balkóne. Každý si v lete nájde trochu viac času na to čo prináša radosť a šťastie ♥.

Čas na sťahovanie z intráku (23 bodov, backtracking)

Skôr ako začne leto, treba sa odsťahovať z internátu a všetky veci pobaliť do krabíc. Ale ako krabice správne naplniť? Ktoré predmety sa do nich zmestia?

Veci budeme baliť do krabíc. Máme zoznam vecí na internáte. O každej veci poznáme jej hmotnosť (kilogramy), objem (litre) a cenu (eurá). O krabici poznáme jej maximálnu nosnosť (kilogramy) a objem (litre). Formát vstupu si zvolíte podľa vlastných preferencií.



Zdroj: Privátny archív z Erasmu

Úlohy: (5b) Vytvorte program, ktorý pre zadané veci a parametre krabice vráti maximálnu cenu vecí, ktoré je možné vložiť do jednej krabice.
(+3b) Vypíšte zoznam vecí, ktoré majú byť v krabici, aby bola ich celková cena maximálna.

Príklad: Zoznam vecí:

monitor — 4 kg, 3 l, 100 €

hrniec — 2 kg, 4 l, 30 €

notebook — 4 kg, 4 l, 700 €

Krabica má nosnosť 7 kg a objem 7 l. Optimálny výber je notebook.

(8b) Vytvorte program, ktorý určí minimálny počet krabíc, so zadanými parametrami, potrebných na zabalenie všetkých vecí.

Poznámka: V tejto úlohe nás nezaujíma cena predmetov.

(+3b) Vypíšte veci uložené v jednotlivých krabiciach.

Príklad: Koľko minimálne krabíc potrebujeme na zabalenie nasledujúcich vecí ak každá krabica ma nosnosť 17 kg a objem 17 l?

vankúš A — 3 kg, 9 l

vankúš B — 3 kg, 9 l

vankúš C — 3 kg, 9 l

vankúš D — 2 kg, 5 l

vankúš E — 2 kg, 5 l

vankúš F — 2 kg, 5 l

vankúš G — 1 kg, 2 l

vankúš H — 1 kg, 2 l

vankúš I — 1 kg, 2 l

Riešením sú 3 krabice.

(+0 až 4b) Body za optimalizáciu programu. Vyžaduje sa vysvetlenie použitých optimalizácií.

Time for game (11 bodov, greedy)

Kamaráti Dominik, Dávid, Milan a Ďuri si chcú cez prázdniny spolu zahrať svoju obľúbenú hru z detstva – Age of Empires 2. Keďže je každý z nich zaneprázdnený, každý napísal intervaly, počas ktorých má voľno. Intervaly sú zadané vo forme začiatku a dĺžky intervalu, pričom čas je vyjadrený ako počet hodín od začiatku týždňa.

Príklad:

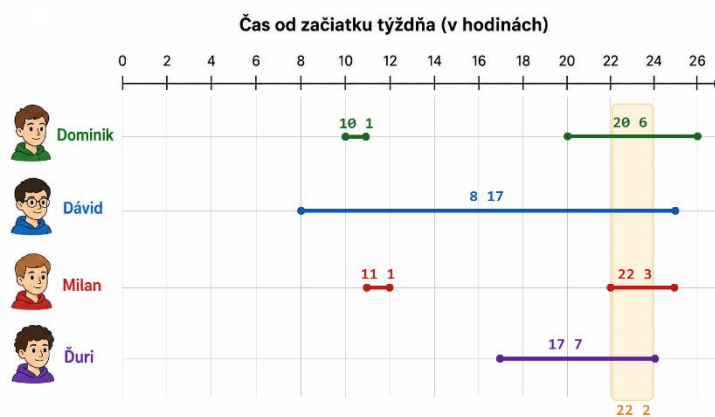
Dominik 10 1, 20 6

Dávid 8 17

Milan 11 1, 22 3

Ďuri 17 7

Riešenie: Áno, môžu sa stretnúť v čase od 22 na 2 hodiny.



Pozn. Ak preferujete iný formát vstupu, informujte o tom dopredu hodnotiteľov. Môžete predpokladať, že intervaly pre jednu osobu sa neprekrývajú, medzi intervalmi je vždy pauza a sú usporiadané chronologicky pre každú osobu.

Úlohy:

(6b) Implementujte program, ktorý načíta vstup a vypíše, či existuje čas, v ktorom sa všetku štyri osoby môžu stretnúť naraz.

(+3b) Nájdite najdlhší súvislý časový úsek, počas ktorého sa môžu všetci stretnúť.

(+2b) Vypíšte konkrétny časový interval, počas ktorého sa môžu stretnúť.

Posledné suroviny (17 bodov, dynamické programovanie)

Dominik, Dávid, Milan a Ďuri si konečne našli čas na hranie. Budovateľské a strategické hry často využívajú suroviny, za ktoré je možné vytvárať jednotky.

Ďuri prišiel o všetky zdroje surovín a nové už nevie získať. Ostalo mu iba drevo a jedlo. Chce využiť všetky zostávajúce suroviny na vytvorenie jednotiek z ponuky n rôznych typov jednotiek, pričom i-ta jednotka stojí $X[i]$ dreva a $Y[i]$ jedla.

Úlohy:

(10b) Napíšte metódu, ktorá pre zoznam jednotiek a zostávajúce množstvá surovín x a y určí, či je možné využiť všetky suroviny. Ľubovoľný počet ľubovoľných jednotiek, cieľom je spotrebovať všetky suroviny (nie maximalizovať počet jednotiek).

Akceptované sú iba riešenia s polynomiálnou časovou zložitosťou (nesmie to byť backtrack ani rekúzia).

(+7b) Ak je možné využiť všetky suroviny, doplňte program tak, aby vypísal jedno možné riešenie, teda aké jednotky a v akom počte má hráč vytvoriť.

Hint: Označme si DP $[i, j]$ hodnotu typu true/false určujúcu, či vieme vytvoriť jednotky za i dreva a j jedla.

Príklad:

Jednotky:

i 0 1 2
 $X = \{0, 2, 4\}$
 $Y = \{8, 4, 2\}$

0	1	2
		
0 	2 	4 
8 	4 	2 

20 
32  = ?

Pre zostatok surovín $x = 20$, $y = 32$ je riešenie áno. 2×0 , 2×1 , 4×2 .

Pizza (20 bodov, grafové algoritmy)

Severín sa celý rok poctivo učil, ale nechce sa cez leto iba flákať. Preto si v malom meste Sobradník našiel brigádu v malej pizzerii na križovatke. Hlavná cesta cez mesto vedie zo severu(north) na juh(south).

Severín si zobral mapu mesta a previedol ju na graf. Vrcholy reprezentujú križovatky a hrany reprezentujú cesty medzi nimi. Váha každej hrany udáva dĺžku cesty v metroch.

Úlohy: (3b) Vytvorte program, ktorý pre zadaný graf a dva vrcholy reprezentujúce severný a južný vstup do mesta vypočíta vzdialenosť medzi nimi.

(+4b) Upravte program tak, aby načítal aj tretí vrchol reprezentujúci pozíciu pizzerie. Program má vrátiť hodnotu true/false podľa toho, či najkratšia cesta medzi severným a južným vstupom do mesta vedie cez pizzeriu.

(+4b) Predpokladajme, že najkratšia cesta z predchádzajúcej úlohy nevedie cez pizzeriu. Zistite, či je možné uzavrieť jednu cestu (túto hranu neuvažujeme) tak, aby najkratšia cesta viedla cez pizzeriu. Ak áno, program vypíše cestu, ktorú treba uzavrieť.

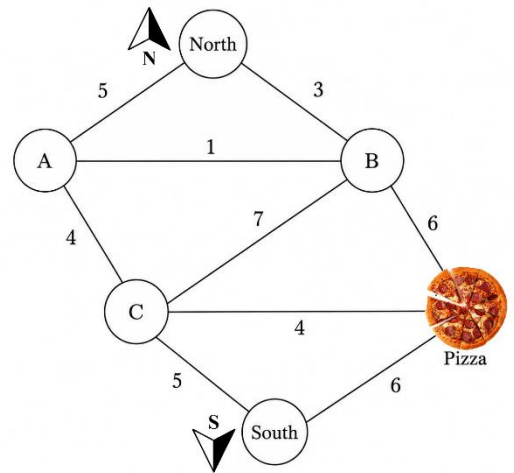
Ďalšie úlohy:

Ked' sa starosta Sobradníka dozvedel, ako Severín pomohol miestnej pizzerii, prišiel za ním so svojím mestským problémom.

Mesto má málo parkovacích miest, a preto chcú niektoré cesty prerobiť na parkovacie zóny. Zároveň však musí zostať zachovaná možnosť dostať sa po zostávajúcich cestách z každej križovatky do každej inej križovatky. Cieľom je vytvoriť čo najviac parkovacích miest. Čím je prerobená cesta dlhšia, tým viac parkovacích miest vznikne.

(6b) Vytvorte program, ktorý pre zadaný graf vypočíta maximálnu celkovú dĺžku ciest, ktoré je možné prerobiť na parkovacie zóny.

(+3b) Vypíšte zoznam ciest, ktoré treba prerobiť na parkovacie zóny.



Zdroj: chatgpt.com, uxwing.com, unsplash.com, manuálne úpravy