

Zadané úlohy

Vybraná skupina: STŘEDNÍK Informatici

Komentář: Soutěž v programování pro studenty prvního ročníku * Kdy: Pátek 30.11.2012 15:00

až 18:00 * Kdo: Studenti 1.ročníku MFF UK oboru informatika * Jak: Skupina v CodExu * Co:

Úlohy, podobně jako ostatní úlohy v CodExu * Přihlašování: Není třeba se přihlašovat, všichni

*budou přihlášení * Submity: povoleno 5 submitů pro každou úlohu * Proč: a) zjistit, - jak na tom*

jsou ti, kdo si myslí, že jim to nejde (a možná to není pravda) - jak na tom jsou ti, kdo si myslí, že

*jim to jde (a možná to není pravda) - kdo má chuť si hrát b) pro radost z řešení problémů **

Výsledky: Až po skončení soutěže * Výhry, kontrola nepodvádění: žádné, každý to dělá pro sebe

Typ skupiny: soukromá | **Vlastník:** Tomáš Holan | **Získané body:** 50

Název úlohy	Data
-------------	------

Body

Omezení

<u>; první</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 11 Max.: 11 Max. 2: 10	Odevzdaných řešení: 1 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; druhá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 11 Max.: 11 Max. 2: 10	Odevzdaných řešení: 2 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; devátá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 0 Max.: 11 Max. 2: 10	Odevzdaných řešení: 0 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; třetí</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 4 Max.: 11 Max. 2: 10	Odevzdaných řešení: 2 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; čtvrtá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 10 Max.: 11 Max. 2: 10	Odevzdaných řešení: 1 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; pátá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 10 Max.: 11 Max. 2: 10	Odevzdaných řešení: 2 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; šestá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 0 Max.: 11 Max. 2: 10	Odevzdaných řešení: 0 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; sedmá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 4 Max.: 11 Max. 2: 10	Odevzdaných řešení: 3 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení
<u>; osmá</u>	Zadána: 30.11. (15h) 1. Termín: 30.11. (17h) 2. Termín: 30.11. (18h)	Přiděleno: 0 Max.: 11 Max. 2: 10	Odevzdaných řešení: 1 z 5 Přípony: pas,c,cpp,cs	Odevzdaná řešení

Celé jméno	; první	; druhá	; devátá	; třetí	; čtvrtá	; pátá	; šestá	; sedmá	; osmá	Body za úlohy	Bonusové body	Body celkem
<i>Lukáš Folwarczný</i>	11	11	9	11	11	11	11	11	10	96	0	96
<i>Jan Hadrava</i>	11	11	-	-	11	11	11	8	8	71	0	71
<i>Matouš Helikar</i>	11	11	-	11	11	11	-	10	6	71	0	71
<i>Tomáš Paladin Volf</i>	11	11	-	10	11	11	-	11	2	67	0	67
<i>Monika Daniláková</i>	11	11	-	11	2	11	10	10	-	66	0	66
<i>Jozef Lelič</i>	11	11	-	4	8	11	11	10	-	66	0	66
<i>Filip Šedivý</i>	11	5	-	11	6	11	11	-	-	55	0	55
<i>Ondřej Švec</i>	11	11	-	4	10	10	-	4	0	50	0	50
<i>Romana Kapustová</i>	11	11	-	-	11	11	-	-	-	44	0	44
<i>Filip Řeha</i>	11	11	-	-	10	11	-	-	-	43	0	43
<i>Jakub Náplava</i>	11	11	-	-	-	10	4	-	-	36	0	36
<i>Petra Kašánková</i>	11	-	-	4	8	11	-	-	-	34	0	34
<i>Ondřej Kubíček</i>	11	11	-	-	-	10	-	-	-	32	0	32
<i>Jan Majerníček</i>	11	10	-	-	-	11	-	-	-	32	0	32
<i>Ladislav Maleček</i>	11	11	-	-	-	8	2	-	-	32	0	32
<i>Martin Mirbauer</i>	11	-	-	-	-	11	-	10	-	32	0	32
<i>Michal Berg</i>	11	1	2	-	-	11	-	5	-	30	0	30
<i>Šimon Rozsival</i>	11	11	-	-	8	0	0	-	-	30	0	30
<i>Šimon Schagerer</i>	11	1	-	-	-	11	-	-	-	23	0	23
<i>Josefina Mádrová</i>	11	5	-	-	-	6	-	-	-	22	0	22
<i>Michal Navrátil</i>	11	-	-	-	-	10	-	-	-	21	0	21
<i>Martin Robovský</i>	11	0	-	-	-	10	-	-	-	21	0	21
<i>Michal Anděra</i>	10	10	-	-	-	-	-	-	-	20	0	20
<i>Jan Klůj</i>	11	1	-	-	0	8	-	-	-	20	0	20
<i>Andrei Yakubouski</i>	11	-	-	-	-	-	-	6	-	17	0	17
<i>František Ember</i>	11	1	-	-	-	-	-	-	-	12	0	12
<i>Eliška Pilátová</i>	11	-	-	-	-	-	-	-	-	11	0	11
<i>Pavel Kratochvíl</i>	-	-	-	-	-	-	8	-	-	8	0	8

Vybraná zadaná úloha: ; první

Autor: Tomáš Holan | Poslední změna: 27.11. (16h) | Zadána: 30.11. (15h) | 1.

Termín: 30.11. (17h) (proběhl) | 2. Termín: 30.11. (18h) (proběhl) | Bodový limit: 11 / 10 / 0 | Získané body: 11 | Max. odevzdaných řešení: 5 (zbývá 4)

Napište program, který na standardní výstup vypíše středník (",").

Vybraná zadaná úloha: ; druhá

Autor: Tomáš Holan | **Poslední změna:** 27.11. (16h) | **Zadána:** 30.11. (15h) | **1.**

Termín: 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 11 | **Max. odevzdaných řešení:** 5 (zbývá 3)

Program přečte ze vstupu tři řádky:

- na prvním řádku je řetězec CO
- na druhém řádku je řetězec ČÍM
- na třetím řádku je řetězec KDE

Řetězce CO, ČÍM a KDE jsou dlouhé nejvýše 255 znaků.

Vytiskne řetězec vzniklý tak, že v řetězci KDE každý výskyt řetězce CO nahradí řetězcem ČÍM. Můžete předpokládat, že se výskyty řetězce CO v řetězci KDE nebudou vzájemně překrývat.

Příklad:

Vstup:

A

B

ABECEDA

Výstup:

BBECEDB

Vybraná zadaná úloha: ; třetí

Autor: Pavel Töpfer | **Poslední změna:** 27.11. (16:01) | **Zadána:** 30.11. (15h) | **1.**

Termín: 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 4 | **Max. odevzdaných řešení:** 5 (zbývá 3)

Ve velké garáži je N parkovacích míst, která jsou očíslována od 1 do N včetně. Garáž otvírá každé ráno prázdná. Kdykoliv do garáže přijede auto, hlídač zkontroluje, zda je některé parkovací místo volné. Pokud není, auto musí počkat u vjezdu do garáže, dokud se nějaké místo neuvolní. Je-li v garáži více volných míst, auto zaparkuje na volném místě s nejmenším číslem. Jestliže do garáže přijedou další auta v době, kdy nějaké auto čeká na uvolnění místa, všechna tato auta se řadí u vjezdu do fronty v tom pořadí, v jakém přijela. Když se pak uvolní nějaké parkovací místo, zaparkuje na něm první auto z fronty (tzn. to z čekajících aut, které přijelo nejdříve). Cena za parkování se počítá jako součin váhy auta a koeficientu, jehož hodnota závisí na použitém parkovacím místě. Cena nezávisí na tom, jak dlouho auto zůstane v garáži.

Napište program, který na základě znalosti cenových koeficientů jednotlivých parkovacích míst v garáži a údajů o přijíždějících autech (váhy aut, pořadí jejich příjezdů a odjezdů) určí celkovou částku, kterou řidiči parkujících aut zaplatí během jednoho dne.

Program dostane na standardním vstupu následující údaje: První řádek obsahuje celá čísla N (počet parkovacích míst, $N \leq 100$) a M (počet parkujících aut, $M \leq 2000$) oddělená mezerou. Další N řádků udává cenové koeficienty parkovacích míst (v pořadí s -tý z těchto řádků obsahuje jedno celé číslo - cenový koeficient parkovacího místa číslo s). Následujících M řádků určuje váhy aut. Auta jsou očíslována od 1 do M včetně, a to bez ohledu na pořadí jejich příjezdu do garáže nebo odjezdu. V pořadí k -tý z těchto M řádků obsahuje jedno celé číslo - váhu auta číslo k). Posledních $2 \cdot M$ řádků popisuje chronologicky, jak probíhají příjezdy a odjezdy jednotlivých aut. Kladné celé číslo i znamená, že auto číslo i přijíždí do garáže. Záporné celé číslo $-i$ znamená, že auto číslo i odjíždí z garáže. Žádné auto nebude odjíždět z garáže dříve, než přijelo, a číslo každého auta od 1 do M se objeví v této posloupnosti právě dvakrát – jednou při příjezdu a podruhé při odjezdu. Žádné auto neodjede z garáže před zaparkováním (tzn. žádné auto neopustí předčasně frontu, v níž čeká na parkování).

Program vypíše na standardní výstup jeden řádek obsahující jedno celé číslo – celkovou částku, kterou všechna parkující auta zaplatí.

Příklad vstupu:

```
3 4
2
3
5
200
100
300
800
3
2
-3
1
4
-4
-2
-1
```

Odpovídající výstup:

5300

Vysvětlení:

Auto číslo 3 zaparkuje na místě 1 a zaplatí $300 \cdot 2 = 600$.

Auto číslo 2 zaparkuje na místě 2 a zaplatí $100 \cdot 3 = 300$.

Auto číslo 1 zaparkuje na místě 1 (které se uvolnilo po odjezdu auta číslo 3) a zaplatí $200 \cdot 2 = 400$.

Auto číslo 4 zaparkuje na místě 3 (poslední zbývajícím místem) a zaplatí $800 \cdot 5 = 4\,000$.

Vybraná zadaná úloha: ; čtvrtá

Autor: Tomáš Holan | **Poslední změna:** 27.11. (16:01) | **Zadána:** 30.11. (15h) | **1. Termín:** 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 10 | **Max. odevzdaných řešení:** 5 (zbývá 4)

Obdélník nebo čtverec o N sloupcích a M řádcích můžeme vyplnit čísly od 1 do $N \cdot M$ tak, že do začneme čísla vepisovat odleva do prvního řádku a dál postupujeme po obvodu po směru hodinových ručiček, například

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

Program přečte ze vstupu čtyři čísla N, M, i a j (z rozsahu 1..10000) a vypíše hodnotu, která v takto vyplněném obdélníku o N sloupcích a M řádcích bude v j-tém řádku i-tého sloupce.

Například pro vstup

5 5 2 4

bude výstup

23

Vybraná zadaná úloha: ; pátá

Autor: Tomáš Holan | Poslední změna: 27.11. (16:01) | Zadána: 30.11. (15h) | 1.

Termín: 30.11. (17h) (proběhl) | 2. Termín: 30.11. (18h) (proběhl) | Bodový limit: 11 / 10 / 0 | Získané body: 10 | Max. odevzdaných řešení: 5 (zbývá 3)

Program přečte ze vstupu celé číslo v rozsahu 0..65535 (typ word, 16 bitů) a vytiskne číslo vzniklé převrácením pořadí jednotlivých bitů s ignorováním počátečních nul.

Například pro vstup 16 = $(10000)_2$

vytiskne hodnotu 1 = $(00001)_2 = 1$

pro vstup 27 = $(11011)_2$

vytiskne hodnotu 27 = $(11011)_2$

Vybraná zadaná úloha: ; šestá

Autor: Pavel Töpfer | **Poslední změna:** 27.11. (16:02) | **Zadána:** 30.11. (15h) | **1. Termín:** 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 0 | **Max. odevzdaných řešení:** 5 (zbývá 5)

Alfons chce ve stánku prodávat párky v rohlíku. Nechal si udělat průzkum, kolik korun jsou lidé ochotni za jeden párek v rohlíku zaplatit. Získal rozsáhlý soubor celých čísel - každé z nich udává, při jaké maximální ceně dotazovaný člověk párek v rohlíku ještě koupí. Alfons chce stanovit prodejní cenu tak, aby utržil co nejvíce. Bude-li cena příliš nízká, koupí si párek téměř každý, ale příjem z každého prodaného párku bude nízký. Bude-li cena příliš vysoká, koupí si párek málo lidí, takže celková Alfonsova tržba bude rovně nízká.

Vstupem programu je textový soubor 'vstup.txt' obsahující celá čísla z rozmezí od 1 do 1000, oddělená od sebe mezerami. Každé z nich určuje odpověď jednoho zákazníka na otázku, za jakou maximální cenu si koupí párek v rohlíku. Čísla v souboru nejsou nijak uspořádána, hodnoty se mohou opakovat. Vstupní data jsou zakončena číslem 0. Výsledkem je jediné celé číslo zapsané na standardní výstup - prodejní cena jednoho párku v rohlíku, při níž dosáhne Alfons maximální celkové tržby. Pokud lze dosáhnout maximální tržby při více různých cenách, program vypíše nejnížší z nich.

Příklad vstupního souboru 'vstup.txt':

```
20 20 15 20 23 8 15 20 25 23 0
```

Odpovídající výstup:

```
20
```


Vybraná zadaná úloha: ; sedmá

Autor: Pavel Töpfer | **Poslední změna:** 27.11. (16:02) | **Zadána:** 30.11. (15h) | **1. Termín:** 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 4 | **Max. odevzdaných řešení:** 5 (zbývá 2)

Na prvním řádku standardního vstupu jsou zadána tři kladná celá čísla D, M, R oddělená mezerami. Tato čísla představují datum (den, měsíc, rok) - jedná se o korektní datum z rozsahu platnosti současného Gregoriánského kalendáře. Na druhém řádku vstupu je zadáno jedno kladné celé číslo N. Program určí a vypíše na standardní výstup datum dne, který bude následovat N dní po datu zadaném na vstupu, a to opět ve tvaru tří celých čísel oddělených mezerami.

Poznámky:

Jednotlivé měsíce v roce mají po řadě 31, 28 (nebo 29), 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 dní.

Rok je přestupný, je-li jeho letopočet dělitelný 4, s výjimkou celých století, která přestupná nejsou. Pokud je ovšem číslo roku dělitelné 400, pak je tento rok přestupný, i když se jedná o celé století.

Hodnota čísla N může odpovídat období délky několika set let, takže se nemusí vejít do 2-bytového typu integer, určitě se ale vejde do 4-bytového celočíselného typu.

Příklad vstupu:

```
30 4 2009
366
```

Odpovídající výstup:

```
1 5 2010
```

Vybraná zadaná úloha: ; osmá

Autor: Pavel Töpfer | **Poslední změna:** 27.11. (16:02) | **Zadána:** 30.11. (15h) | **1. Termín:** 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 0 | **Max. odevzdaných řešení:** 5 (zbývá 4)

Na poště jsou tři rovnocenné přepážky. Přicházející zákazníci se před nimi řadí do jedné společné fronty a jsou postupně obsluhováni. Pro každého zákazníka je zadán čas jeho příchodu na poštu (vyjádřeno celočíselně v minutách od okamžiku otevření pošty) a délka obsluhy (rovněž vyjádřeno celočíselně v minutách). Ve vstupním souboru LIDI.TXT jsou zákazníci seřazeni podle doby svého příchodu na poštu, oba údaje o každém zákazníkovi jsou vždy na jednom řádku a jsou odděleny mezerou.

Úkolem programu je zjistit:

a) kdy budou obslouženi všichni zákazníci (celočíselný údaj v minutách, měřeno od otevření pošty)

b) průměrnou dobu čekání zákazníka ve frontě (údaj v minutách, zaokrouhlený na celé minuty).

Oba nalezené údaje program vypíše v uvedeném pořadí na standardní výstup a oddělí je mezerou.

Příklad vstupu:

5 10 6 15 7 12 8 3 10 6

Odpovídající výstup:

24 3

Vybraná zadaná úloha: ; devátá

Autor: Pavel Töpfer | **Poslední změna:** 27.11. (16:02) | **Zadána:** 30.11. (15h) | **1. Termín:** 30.11. (17h) (proběhl) | **2. Termín:** 30.11. (18h) (proběhl) | **Bodový limit:** 11 / 10 / 0 | **Získané body:** 0 | **Max. odevzdaných řešení:** 5 (zbývá 5)

Do soutěže v řešení matematických úloh se přihlásilo N studentů, kteří obdrželi identifikační čísla od 1 do N . V soutěži bylo zadáno T úloh, označených čísly od 1 do T . Počet bodů přidělených za správně vyřešenou úlohu se stanoví až po skončení soutěže a je roven počtu soutěžících, kteří tuto úlohu nevyřešili. Výsledné hodnocení soutěžícího je rovno součtu bodů, které soutěžící získal za všechny jím vyřešené úlohy. Ve výsledkové listině jsou soutěžící uspořádáni sestupně podle dosaženého hodnocení. V případě shodného počtu získaných bodů je ve výsledkové listině umístěn výše ten soutěžící, který vyřešil větší počet úloh. Pokud o vzájemném pořadí soutěžících nerozhodne ani toto kritérium, soutěžící se stejným hodnocením i stejným počtem vyřešených úloh budou uspořádáni vzestupně podle svých identifikačních čísel. Napište program, který pro soutěžícího číslo P určí, kolik získal v soutěži bodů a na kterém místě skončil ve výsledkové listině.

Na prvním řádku vstupního textového souboru 'vstup.txt' jsou uvedena tři kladná celá čísla oddělená mezerami: počet soutěžících N ($N \leq 1000$), počet řešených úloh T ($T \leq 50$), identifikační číslo sledovaného studenta P ($P \leq N$). Dalších N řádků souboru popisuje, které úlohy vyřešil který soutěžící. Tyto řádky jsou upořádány v pořadí podle identifikačních čísel soutěžících. Na každém řádku jsou uvedena čísla úloh, které příslušný soutěžící správně vyřešil - čísla jsou na řádku v libovolném pořadí, bez opakování, oddělená mezerami. Pokud některý soutěžící nevyřešil žádnou úlohu, příslušný řádek bude prázdný. Program vypíše na standardní výstup dvě celá čísla oddělená mezerou: nejprve výsledný počet bodů soutěžícího s identifikačním číslem P a pak jeho pořadové číslo (umístění) ve výsledkové listině.

Příklad vstupního souboru 'vstup.txt':

```
5 3 2
3
1 2
1
2 1
1 2
```

Odpovídající výstup:

```
3 2
```

Vysvětlení:

První úlohu nevyřešil jenom jeden soutěžící, takže bude hodnocena 1 bodem. Druhou úlohu nevyřešili dva soutěžící, bude tedy hodnocena 2 body. Třetí úlohu nevyřešili čtyři soutěžící, takže je hodnocena 4 body. První soutěžící tudíž získal 4 body, druhý soutěžící (což je náš P), čtvrtý a pátý mají všichni po 3 bodech, třetí soutěžící má 1 bod. Soutěžící s identifikačními čísly 2, 4 a 5 mají nejen shodný počet bodů, ale i stejný počet vyřešených úloh, takže jejich vzájemné pořadí bude určeno podle identifikačních čísel. Soutěžící s číslem 2 bude proto ve výsledkové listině druhý.