

PRAKTIKUM IZ PROGRAMIRANJA 2

- domaći zadatak broj 2 – - specijalna grupa -

Cilj zadataka u specijalnoj grupi je da pripremi studente za rešavanje određenih tipova algoritamskih zadataka. Zadaci su problemskog, „takmičarskog“ tipa, što od svakog studenta zahteva odlično poznavanje samog programskog jezika C, kao i osnovnih algoritama i struktura podataka čije korišćenje može pomoći prilikom rešavanja zadataka. Zadaci se rešavaju **samostalno**.

Tema drugog domaćeg zadatka je **dinamičko programiranje (dynamic programming)**. Zavisno od rednog broja problema, sastaviti **jedan** od programa koji se nalaze u prilogu ovog dokumenta. Prilikom rešavanja zadatka voditi računa o zadatim ograničenjima. Poznato je da se svi zadati problemi mogu rešiti dinamičkim programiranjem, te tu činjenicu uzeti u obzir prilikom rešavanja zadatka. Studenti treba da optimizuju svoja rešenja i da na odbranu domaćih zadataka donesu rešenja sa što manjom vremenskom i prostornom složenosti. Tamo gde je to pogodno, koristiti pokazivače i dinamičku alokaciju memorije prilikom alokacije prostora, a ne statičku alokaciju sa maksimalnim definisanim kapacitetom.

Pored osnovnog zadatka, studenti po želji mogu uraditi i **BONUS** zadatak za dodatna tri poena.

Pored dostupnih test primera, odabrati bar tri dodatna skupa podataka sa kojima će program biti testiran. Svaki primer treba da sadrži ulazne podatke i očekivani izlaz za te podatke.

Napomene:

1. Odbrana drugog domaćeg zadatka je u četvrtak, 23.04.2015. po rasporedu dostupnom na sajtu predmeta.
2. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa): **$i = (R + G) \bmod 4$**
3. Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
 - **dz2.c**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku C;
 - **dz2_bonus.c**, koja sadrži izvorni tekst bonus programa na programskom jeziku C, ukoliko se radi i bonus zadatak;

16.04.2015. godine

sa predmeta

0. Restoran „Domovina“

U neka srećnija vremena studenti i profesori ETF-a su svoje uspehe obavezno proslavljali u restoranu „Domovina“ preko puta zgrade fakulteta. Restoran je imao odličan izbor domaće hrane i pića, ali su konobari imali nezgodan običaj da račun naplaćuju zbirno po osobi, bez detaljne specifikacije narudžbe i cena. Profesorima Caji i Bori se nije sviđala ova praksa, pa su zamolili studente računarstva da napišu program koji na osnovu menija restorana „Domovina“ i iznosa računa plaćenog po svakoj osobi rekonstruiše sadržaj svake narudžbe.

Zadatak

Potrebno je pronaći šta je naručeno u svakoj od narudžbi hrane i pića u restoranu, ukoliko su poznati ukupna cena svake narudžbe i pojedinačna cena svake stavke menija. Program treba da za svaku narudžbu ispiše po jednu liniju teksta. Ukoliko postoji jedinstvena narudžba za zadatu ukupnu cenu, ispisati redne brojeve stavki menija koji se sadrže u njoj. Ukoliko zadatoj ukupnoj ceni odgovara više narudžbi različitog sadržaja, ispisati *Ambiguous*. Ukoliko zadatoj ukupnoj ceni ne odgovara nijedna narudžba, ispisati *Impossible*. Narudžba može sadržati više istih stavki.

Opis ulaznih podataka:

Prva linija ulaza sadrži broj N ($1 \leq N \leq 100$) koji predstavlja ukupan broj stavki u meniju restorana. U narednoj liniji se nalazi N celih brojeva C_1, C_2, \dots, C_n koji predstavljaju cenu svake pojedinačne stavke u meniju ($1 \leq C_i \leq 1000$). U trećoj liniji ulaza se nalazi broj M ($1 \leq M \leq 1000$) koji predstavlja broj zadatih narudžbi. U narednoj liniji se nalazi M celih brojeva S_1, S_2, \dots, S_n koji predstavljaju ukupnu cenu svake pojedinačne narudžbe ($1 \leq S_i \leq 1000$).

Očekivani ulaz:

```
3
4 5 8
3
11 13 14
```

Očekivani izlaz:

```
Impossible
Ambiguous
1 2 2
```

Očekivani ulaz:

```
6
215 275 335 355 420 580
1
1505
```

Očekivani izlaz:

```
Ambiguous
```

1. Big pizza

Restoran brze hrane „Big pizza“ želi da proširi svoje poslovanje i stoga žele da emituju reklamu na radiju. Kako se restoran nalazi blizu studentskog doma „Lola“, oni uglavnom žele da privuku studente, pa planiraju da se reklamiraju na internet radiju „Akadamac“ koji slušaju studenti. Radio „Akadamac“ emituje reklame na svakih 15 minuta, a ima odličnu marketinšku službu koja svim zainteresovanim oglašivačima dostavlja podatke slušanosti (broju konekcija) radija prilikom emitovanja svakog reklamnog bloka. Takođe, radio „Akadamac“ zahteva da sve reklame jednog oglašivača budu emitovane u kontinualnom vremenskom periodu u toku jednog dana (npr. u svim reklamnim blokovima od 20 do 22 časa). Istraživanje restorana „Big pizza“ je pokazalo da ukoliko student čuje reklamu, on u proseku potroši jedan evro više na picu. Da bi uvećao svoj profit, restoran „Big pizza“ je angažovao vas da odredite najbolji vremenski period za emitovanje reklama, kako bi maksimizovao svoj profit.

Zadatak

Potrebno je odrediti maksimalni profit koji restoran može da ostvari emitovanjem reklama u određenom kontinualnom vremenskom periodu. Za određeni maksimalni profit, potrebno je ispisati redni broj prvog i poslednjeg reklamnog bloka u kome će reklama biti emitovana.

Opis ulaznih podataka:

Prva linija ulaza sadrži pozitivne cele brojeve N ($1 \leq N \leq 100\,000$) koji predstavlja ukupan broj reklamnih blokova tokom dana i P ($1 \leq P \leq 1000$) koji predstavlja cenu emitovanja reklame u jednom reklamnom bloku. U narednoj liniji se nalazi N celih brojeva L_1, L_2, \dots, L_n koji predstavljaju slušanost radija u toku i -tog reklamnog bloka ($1 \leq L_i \leq 2000$).

Očekivani ulaz:

```
6 20
18 35 6 80 15 21
```

Očekivani izlaz:

```
61
2 4
```

2. Papirići

Perica je oduvek voleo igre na sreću, pa je smislio jednu za svog druga Jovicu. Perica uzme N papirića, na svakom papiriću napiše po jedan broj i okrene ih tako da Jovica ne zna koji su brojevi napisani. Zatim Jovica mora da rasporedi sve papiriće na proizvoljan broj gomila. Svaka gomila mora da ima najmanje 2, a najviše K papirića. Na kraju Perica otkrije sve papiriće, te računa broj poena koji je Jovica osvojio. Perica računa poene na sledeći način: Pronađe najveći i najmanji broj na jednoj gomili i izračuna njihovu razliku. Kada izračuna razliku najvećeg i najmanjeg za svaku gomilu, sabere sve razlike. Dobijena vrednost predstavlja osvojene poene. Što je broj poena manji, to je Jovica uspešniji.

Zadatak

Potrebno je odrediti minimalan broj poena koje Jovica može da sakupi, ukoliko su poznati brojevi koje je Perica napisao na papirićima. Program treba da ispiše minimalan broj poena koje Jovica može da sakupi, kao i raspored papirića po gomilama, za svaku gomilu u zasebnom redu.

Opis ulaznih podataka:

Prva linija ulaza sadrži pozitivne cele brojeve N i K ($2 \leq K \leq N \leq 100\,000$) koji predstavljaju, redom, broj papirića i maksimalnu veličinu gomila. U narednoj liniji se nalazi N celih brojeva, koji predstavljaju brojeve napisane na papirićima.

Očekivani ulaz:

```
7 3
6 0 5 2 8 0 -2
```

Očekivani izlaz:

```
7
0 2
6 5 8
0 -2
```

3. Baba Vanga

Baba Vanga je poznata proročica i želi da napravi sebi magičnu ogrlicu. Njoj je na raspolaganju određeni broj amajlija od kojih može napraviti ogrlicu, a da pritom svaku amajliju iskoristi samo jednom. Svaka amajlija ima određenu težinu, a ukupna težina ogrlice koju baba Vanga može da nosi je ograničena. Takođe, svaka amajlija poseduje određenu magičnu moć, tako da baba Vanga ne želi podjednako sve amajlije u okviru svoje ogrlice, već bi želela da napravi ogrlicu sa maksimalnom magičnom moći.

Zadatak

Ukoliko je poznato da ogrlica ne sme težiti više od M grama, potrebno je odrediti maksimalnu magičnu moć ogrlice, kao i spisak amajlija koje čine ogrlicu, koja bi se mogla sastaviti od dostupnih amajlija. Svaka amajlija se može iskoristiti samo jednom prilikom pravljenja ogrlice.

Opis ulaznih podataka:

Prva linija ulaza sadrži pozitivne cele brojeve N ($2 \leq N \leq 1\,000$) koji predstavlja broj dostupnih amajlija i M ($1 \leq M \leq 10\,000$) koji predstavlja maksimalnu težinu ogrlice. U narednih N linija se nalaze dva cela broja W_i ($1 \leq W_i \leq 400$) i P_i ($1 \leq P_i \leq 100$) koji predstavljaju težinu i magičnu moć i -te amajlije.

Očekivani ulaz:

```
4 6
1 4
2 6
3 12
2 7
```

Očekivani izlaz:

```
23
1 3 4
```

4. Med i orasi (BONUS ZADATAK)

Prijatelj malog Z.-a često posećuje svoju devojku Zitu Arok. Ona ima običaj da ga hrani medom i orasima. Ovog puta se njegova devojka malo naljutila i ne želi da prijatelju malog Z.-a priredi gastronomski užitek. Da bi je ponovo osvojio i dobio omiljenu hranu, Zita Arok je smislila interesantnu igru. Zita Arok kaže prijatelju malog Z.-a koliko oraha želi da mu da, a on treba da odgovori koliko različitih reči se može sastaviti od zadatog broja oraha. Pritom, Zita Arok ne voli podjednako sva slova, već svako slovo vredi određenu količinu oraha. Prijatelj malog Z. ima punu slobodu da sastavlja reči koje god želi. Na primer, reči kao sto su 'aaaaaaaaa', 'mmmmmmm', 'uuuuuu', 'hrewoifrw', njegovoj devojci sasvim odgovaraju, dok god daju odgovor na njeno pitanje. Posto prijatelj malog Z. misli od drugim stvarima i nema vremena za kombinatoriku, on se obratio malom Z.-u da mu pomogne da odgovori na pitanje svoje devojke.

Zadatak

Pretpostavimo da u nekom zadatom alfabetu svaka reč košta određen broj oraha. Pod pretpostavkom da na raspolaganju imamo M oraha, koliko reči možemo sastaviti koje koštaju tačno M oraha? Program treba da odredi i ispiše ukupan broj reči koje je moguće napraviti od zadatih slova koje imaju zadatu cenu koštanja. Smatrati da su sva slova raličita.

Opis ulaznih podataka:

Prva linija ulaza sadrži pozitivne cele brojeve N ($1 \leq N \leq 26$) koji predstavlja broj slova koje mali Z. ima na raspolaganju i M ($1 \leq M \leq 2^{30}$) koji predstavlja broj oraha na raspolaganju. U narednoj liniji se nalazi N celih brojeva broj X_i ($1 \leq X_i \leq N$), gde X_i predstavlja vrednost i-tog slova u broju oraha.

Očekivani ulaz:

```
26 1
23 20 1 15 2 10 21 18 11 6 1 10 1 7 8 13 26 13 16 26 21 24 5 3 13 15
```

Očekivani izlaz:

```
3
```

Objašnjenje:

Imamo tri slova koja mogu da se sastave od jednog oraha i samo jedan orah na raspolaganju, što znači da možemo da sastavimo tri različite reči (reč može biti i dužine jedan).