

PRAKTIKUM IZ PROGRAMIRANJA 2

- domaći zadatak broj 2 –

- specijalna grupa -

Cilj zadataka u specijalnoj grupi je priprema studenata 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. 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 složenošću. Pored osnovnog zadatka, studenti po želji mogu uraditi i **BONUS** zadatak koji ima dodatnu težinu.

Urađen domaći zadatak u specijalnoj grupi će biti ocenjen sa 2 poena koja će se računati kao dodatni poeni u odnosu na poene osvojene na redovnim domaćim zadacima. Urađen BONUS domaći zadatak u specijalnoj grupi će biti ocenjen sa 1 poen koji će se računati kao dodatni poeni u odnosu na poene osvojene na redovnim domaćim zadacima. Planirano je održavanje dva domaća zadatka u specijalnoj grupi.

Pored odbrane pred demonstratorima, za testiranje rešenja zadataka biće korišćen virtualni *online judge system* (<https://vjudge.net>). Uz tekst svakog domaćeg zadatka je dat link ka originalnom problemu putem kojeg se može izvršiti predaja zadatka i testiranje na sistemu.

Studenti koji žele da rade domaće zadatke treba da se registruju na ovaj sistem i pridruže odgovarajućoj grupi. Više detalja će biti saopšteno na predmetnoj listi elektronske pošte.

Napomene:

1. Odbrana domaćih zadataka u specijalnoj grupi će se obaviti nakon što se normalizuje proces rada na fakultetu, kada se za to steknu mogućnosti.
2. Rok za izradu i predaju domaćeg zadatka na online judge platformi je **ponedeljak, 25.05.2020. u 23:59**. Zadaci postavljeni nakon tog roka neće biti uzeti u obzir prilikom pregledanja.
3. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa): **$i = (R + G) \bmod 4$**
4. Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
 - **dz2.c**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku C;
 - **dz2_mod.c**, koja sadrži izvorni tekst bonus programa na programskom jeziku C;

28.04.2020. godine

Sa predmeta

0. Crno i belo

Dat je niz celih brojeva. Dozvoljeno je obojiti svaki od brojeva u belu ili crnu boju ili ostaviti broj neobojen. Niz brojeva ofarbanih u crnu boju treba da bude u strogo rastućem poretku, dok niz brojeva ofarbanih u belu boju treba da bude u strogo opadajućem poretku. Vaš zadatak je da minimizujete broj neobojenih brojeva.

Opis ulaznih podataka

Ulaz sadrži više test primera.

Svaki test primer sadrži dva reda. U prvom redu se nalazi broj elemenata u nizu N ($1 \leq N \leq 200$).

U sledećem redu se nalazi N celih brojeva X_i odvojenih razmakom, koji predstavljaju niz brojeva ($1 \leq X_i \leq 10^6$).

Nakon poslednjeg test primera, sledi red koji sadrži samo broj -1.

Opis izlaznog fajla:

Za svaki test primer potrebno je ispisati minimalni broj neofarbanih brojeva, svaki u zasebnom redu.

Primeri

Ulaz:

```
8
1 4 2 3 3 2 4 1
12
7 8 1 2 4 6 3 5 2 1 8 7
-1
```

Izlaz:

```
0
2
```

Link ka originalnom zadatku:

<https://vjudge.net/contest/368294#problem/A>

1. OktoberCity

U OktobarCity gradu se sve kuće nalaze u jednoj ulici. Svaka kuća ima svoju visinu. Potrebno je pronaći maksimalnu moguću podsekvencu ovih kuća gde će važiti sledeće pravilo: visine kuća u podsekvenci su prvo strogo rastuće, a onda nakon toga strogo opadajuće.

Opis ulaznih podataka

U prvom redu se nalazi ceo broj T ($T < 100$) – ukupan broj test primera. Svaki test primer sadrži dva reda. U prvom redu se nalazi broj kuća u gradu N ($1 \leq N \leq 1000$). U sledećem redu se nalazi N celih brojeva A_i odvojenih razmakom, koji predstavljaju visine kuća ($1 \leq A_i \leq 1000$).

Opis izlaznog fajla:

Za svaki test primer potrebno je ispisati maksimalnu dužinu podsekvence kuća, svaki u zasebnom redu.

Primeri

Ulaz:

```
2
10
1 3 5 6 4 8 4 3 2 1
6
8 6 3 4 2 1
```

Izlaz:

```
9
5
```

Link ka originalnom zadatku:

<https://vjudge.net/contest/368294#problem/B>

2. Kupovina jabuka

Milica je otišla u supermarket da kupi tačno k kilograma jabuka za svojih n prijatelja. Supermarket u koji je otišla je čudan. Jabuke se ne prodaju na kilogram, već su spakovane u džakove koji sadrže po k kg jabuka, gde je k pozitivan ceo broj. Cena džakova sa jabukama je data u tabeli, gde vrednost k -tog polja u tabeli predstavlja cenu džaka sa k kg jabuka. Ako je vrednost k -tog polja u tabeli jednaka -1, to znači da paket sa k kg jabuka ne postoji na stanju. Milica je odlučila da kupi najviše n džakova sa jabukama. Koliko najmanje novca Milica mora da potroši?

Opis ulaznih podataka:

U prvom redu se nalazi ceo broj C ($1 \leq C \leq 1000$) – ukupan broj test primera.

Svaki test primer sadrži dva reda. U prvom redu se nalaze N i K , broj prijatelja i ukupan broj kilograma jabuka koje je potrebno kupiti. ($1 \leq N \leq 100, 1 \leq K \leq 100$).

U sledećem redu se nalazi K celih brojeva X_i odvojenih razmakom, koji predstavljaju cenu džaka sa i kg jabuka ili -1 u slučaju da takvih džakova nema na stanju ($1 \leq X_i \leq 1000$).

Opis izlaznog fajla:

Za svaki test primer potrebno je ispisati minimalni iznos koji je potrebno potrošiti ili -1 ako nije moguće ispuniti zadatak, svaki u zasebnom redu.

Primeri

Ulaz:

```
2
3 5
-1 -1 4 5 -1
5 5
1 2 3 4 5
```

Izlaz:

```
-1
5
```

Link ka originalnom zadatku:

<https://vjudge.net/contest/368294#problem/C>

3. Alternativna podsekvenc

Dat je niz celih brojeva a_1, a_2, \dots, a_n . Potrebno je naći najdužu moguću alternativnu podsekvencu. Sekvenc brojeva b_1, b_2, \dots, b_k je alternativna ako važi:

1. $|b_1| < |b_2| < \dots < |b_k|$
2. Susjedni elementi u nizu su alternativnih znakova.

Opis ulaznih podataka:

U prvom redu se nalazi ceo broj N ($1 \leq N \leq 5000$) – ukupan broj elemenata u nizu.

U sledećem redu se nalazi N celih brojeva a_i odvojenih razmakom, koji predstavljaju brojeve u nizu ($1 \leq |a_i| \leq 10^9$).

Opis izlaznog fajla:

Potrebno je ispisati dužinu najduže moguće alternativne podsekvence.

Primeri

Ulaz:

8
1 2 -2 -3 5 -7 -8 10

Izlaz:

5

Link ka originalnom zadatku:

<https://vjudge.net/contest/368294#problem/D>

4. Modifikacija niza (BONUS ZADATAK)

Dat je niz celih brojeva a_1, a_2, \dots, a_n . Nad nizom je moguće raditi dve vrste modifikacije niza: dodavanje elementa niza na bilo koje mesto u nizu (uključujući i početak i kraj niza) i izbacivanje bilo kog elementa iz niza.

Kao rezultat potrebno je dobiti niz koji je moguće podeliti u podnizove koji predstavljaju permutacije. Svaki element niza mora da pripada isključivo jednom podnizu. Svaki podniz mora da sadrži sve brojeve od 1 do veličine tog podniza, svaki po jednom.

Potrebno je pronaći koliko je najmanje modifikacija potrebno izvršiti tako da se dobije traženi rezultat.

Opis ulaznih podataka:

U prvom redu se nalazi ceo broj T ($T < 1000$) – ukupan broj test primera.

Svaki test primer sadrži dva reda. U prvom redu se nalazi broj elemenata u nizu N ($1 \leq N \leq 10^6$).

U sledećem redu se nalazi N celih brojeva A_i odvojenih razmakom, koji predstavljaju niz elemenata ($1 \leq A_i \leq 10^9$).

Opis izlaznog fajla:

Za svaki test primer potrebno je ispisati minimalan broj modifikacija koje je potrebno izvršiti, svaki u zasebnom redu.

Primeri

Ulaz:

```
2
5
1 4 1 2 2
4
2 3 2 3
```

Izlaz:

```
1
2
```

Link ka originalnom zadatku:

<https://vjudge.net/contest/368294#problem/E>