

Zadaća 1.

Ova zadaća nosi ukupno 4 poena (od 20 poena, koliko će ukupno nositi sve zadaće zajedno), pri čemu svaki zadatak nosi po 0,8 poena. Svi zadaci se mogu uraditi na osnovu gradiva sa prva četiri predavanja i pretpostavljenog predznanja iz predmeta "Osnove računarstva". Rok za predaju ove zadaće je utorak, 27. III 2012. (do kraja dana) i ne može se produžiti. Zadaće se predaju putem Zamgera.

1. U nekom deseterospratnom skladištu nalazi se lift za prevoz tereta koji se upravlja uz pomoć računarskog programa. Trenutna pozicija lifta čuva se u promjenljivoj "sprat" čije su legalne vrijednosti u opsegu od 0 do 10 (0 odgovara prizemlju). Trenutni smjer kretanja lifta (tj. smjer u kojem će se lift kretati nakon što se pokrene) čuva se u promjenljivoj "smjer". Ova promjenljiva je tipa "Smjerovi", koji predstavlja pobrojani tip definiran kao

```
enum Smjerovi {Nagore, Nadolje};
```

Potrebno je napraviti program koji korisniku nudi sljedeće komande za upravljanje liftom: I – Idi; P – Promijeni smjer; K – Kraj. Komanda I obavezno je praćena jednim nenegativnim cijelim brojem (npr. 15) i ona pomjera lift za navedeni broj spratova u smjeru u kojem je lift trenutno usmjeren. U slučaju da je nemoguće pomjerenje za zadani broj spratova, treba prijaviti poruku o greški. Komanda P mijenja smjer kretanja lifta, odnosno ukoliko se lift trenutno kreće nagore, nakon ove komande kretaće se nadolje, i obrnuto. Komanda K završava program. Sve druge komande su ilegalne, i trebaju dovesti do prijave greške i ponovnog izbora komande. Na početku rada, lift se nalazi u prizemlju i kreće se nagore. Dijalog između programa i korisnika mogao bi izgledati poput sljedećeg:

```
Lift se nalazi u prizemlju i kreće se nagore.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I
Pogrešna komanda (nedostaje parametar)!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I5
Lift se nalazi na 5. spratu i kreće se nagore.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I1
Lift se nalazi na 6. spratu i kreće se nagore.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): P
Lift se nalazi na 6. spratu i kreće se nadolje.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): S
Pogrešna komanda!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): IXY2
Pogrešna komanda (neispravan parametar)!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I2XY
Pogrešna komanda (neispravan parametar)!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I0
Lift se nalazi na 6. spratu i kreće se nadolje.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I-1
Pogrešna komanda (neispravan parametar)!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I8
Nije moguće pomjerenje za navedeni iznos spratova!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I2
Lift se nalazi na 4. spratu i kreće se nadolje.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): I4
Lift se nalazi u prizemlju i kreće se nadolje.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): P2
Pogrešna komanda (suvišan parametar)!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): P
Lift se nalazi na 4. spratu i kreće se nagore.
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): KK
Pogrešna komanda (suvišan parametar)!
Unesi komandu (I - Idi, P - Promijeni smjer, K - Kraj): K
Dovidjenja!
```



Naravno, konkretan dijalog zavisi od toga šta je korisnik unio, ali poruke koje se ispisuju na ekranu trebaju izgledati tačno kao što je u ovom dijalogu prikazano. Budite sigurno da ste predvidjeli ispravne reakcije na bilo šta što bi eventualno mogao unijeti korisnik (program ne smije da “crkne” ma šta korisnik unio). Napomena: funkcija “peek” će Vam vjerovatno biti od velike koristi.

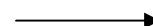
2. Napišite funkciju “SlikeUOgledalu” koja kao parametar prima vektor cijelih brojeva a koja kao rezultat vraća novi vektor koji sadrži brojeve koji su “slika u ogledalu” odgovarajućih brojeva iz prvog vektora, odnosno koji se od odgovarajućih brojeva iz prvog vektora razlikuju po tome što im je redoslijed cifara obrnut. Na primjer, ukoliko se funkciji prosljedi vektor čiji su elementi redom 327, -42, 5, 46127, -319 i 87, vektor koji se vraća kao rezultat treba redom da sadrži elemente 723, -24, 5, 72164, -913 i 78. Napisanu funkciju demonstrirajte u testnom programu koji traži da se sa tastature unose brojevi koji se smještaju u neki vektor sve dok se sa tastature ne unese nula (koja se ne smješta u vektor), a koji nakon toga poziva napisanu funkciju da odredi “slike u ogledalu” unesenih brojeva i ispiše ih na ekran u vidu rečenica oblika “Slika u ogledalu broja x glasi y ” i tako redom za sve brojeve. U programu ne treba testirati ispravnost unesenih podataka, tj. pretpostavićemo da će korisnik programa zaista unositi brojeve. Napomena: nikakav ispis ne treba vršiti unutar funkcije, nego samo u glavnom programu!
3. Napišite funkciju “BrojRijeci” koja kao parametar prima jedan string (tj. parametar tipa “string”) koji predstavlja neku rečenicu, a koja vraća kao rezultat broj riječi unutar te rečenice. Pri tome se pod pojmom “riječ” podrazumijeva svaka skupina susjednih znakova među kojima nema razmaka, takva da se ispred prvog znaka nalazi razmak (osim ukoliko se prvi znak nalazi na samom početku stringa) i da se iza posljednjeg znaka nalazi razmak (osim ukoliko se posljednji znak nalazi na samom kraju stringa). Na primjer, string “Ovo je primjer.” sastoji se od tri riječi, a isto vrijedi i za string “ Ovo je primjer. ”. Napisanu funkciju testirajte u glavnom programu na stringovima koji se unose sa tastature.
4. U teoriji brojeva često se javlja potreba da se neki prirodan broj n prikaže u obliku $n = p q^2$ gdje su p i q prirodni brojevi takvi da p ima sve proste faktore različite (tj. u rastavi broja p na proste faktore niti jedan prosti faktor ne javlja se više puta, odnosno sa eksponentom većim od 1). Na primjer, imamo $28586250 = 42 \cdot 825^2$, pri čemu 42 ima sve proste faktore različite (zaista, imamo $42 = 2 \cdot 3 \cdot 7$). Ovakav prikaz nije teško dobiti nakon što prethodno rastavimo broj n na proste faktore. Na primjer, imamo ovakav račun:

$$\begin{aligned} 28586250 &= 2 \cdot 3^3 \cdot 5^4 \cdot 7 \cdot 11^2 = 2 \cdot 3 \cdot 3^2 \cdot 5^4 \cdot 7 \cdot 11^2 = (2 \cdot 3 \cdot 7) \cdot (3^2 \cdot 5^4 \cdot 11^2) = \\ &= (2 \cdot 3 \cdot 7) \cdot (3 \cdot 5^2 \cdot 11)^2 = 42 \cdot 825^2 \end{aligned}$$

Generalizirajte ovaj primjer tako što ćete napraviti funkciju “RastavaBroja” sa tri parametra “n”, “fmin” i “fmax” koja za zadani prirodan broj n (koji se zadaje kao prvi parametar “n” ove funkcije) pronalazi rastavu oblika $n = p q^2$ pri čemu p zadovoljava postavljene uvjete i nadene vrijednosti p i q smješta respektivno u parametre “p” i “q”. Na primjer, ukoliko se kao prvi parametar zada broj 28586250, u drugi i treći parametar treba da se smjeste respektivno brojevi 42 i 825. U slučaju da n nije prirodan broj, funkcija treba baciti izuzetak. Funkcija se u svom radu ne treba oslanjati ni na kakve druge funkcije. Napisanu funkciju demonstrirajte u testnom programu u kojem se za broj unesen sa tastature ispisuje odgovarajuće brojeve p i q . Program treba da testira da li je korisnik zaista unio broj i da prijavi odgovarajuću poruku o greški ukoliko to nije slučaj. Obavezno predvidite i hvatanje eventualno bačenog izuzetka.

Napomena 1: prosti brojevi su, za onog ko to ne zna (a praksa pokazuje da ima dosta onih koji to ne znaju), oni brojevi koji imaju tačno dva djelioca, jedinicu i samog sebe. Prosti faktori nekog broja su oni njegovi faktori koji su prosti brojevi.

Napomena 2: Dozvoljeno je u nekim slučajevima da bude $p = 1$ (to će se desiti kad god je n kvadrat nekog cijelog broja, npr. $15876 = 1 \cdot 126^2$) mada, strogo rečeno, broj 1 nema prostih faktora pa je malo diskutabilno reći da su “svi njegovi prosti faktori različiti”.



5. Neki element matrice naziva se njen *lokalni minimum* ukoliko je on striktno manji od svih ostalih elemenata koji se nalaze u istom redu ili u istoj koloni kao i taj element. Analogno se definira i *lokalni maksimum*. Može se desiti da matrica uopće nema lokalnih minimuma ili maksimuma, a može se desiti da ih ima više od jednog. Na primjer, sljedeća matrica ima tri lokalna minimuma (to su elementi na pozicijama (1,3), (2,4) i (4,1)) i dva lokalna maksimuma (to su elementi na pozicijama (1,5) i (2,2)):

$$\begin{pmatrix} 10 & 7 & 2 & 5 & 14 \\ 9 & 12 & 6 & 3 & 5 \\ 13 & 6 & 11 & 4 & 13 \\ 2 & 5 & 3 & 7 & 9 \end{pmatrix}$$

Napišite funkciju “EkstremiMatrice” koja kao prvi parametar prima matricu realnih brojeva organiziranu kao vektor vektora realnih brojeva, a koja u drugi i treći parametar smješta respektivno broj lokalnih minimuma i broj lokalnih maksimuma matrice. Ukoliko preneseni parametar nema strukturu matrice (tj. ukoliko svi redovi proslijeđenog vektora vektora nemaju isti broj elemenata), funkcija treba baciti izuzetak. Napisanu funkciju testirajte u glavnom programu na primjeru matrice čije se dimenzije i elementi unose sa tastature. Pri tome ćemo pretpostaviti da će uneseni podaci biti korektni, odnosno ne treba testirati ispravnost unesenih podataka. Napomena: u ovom konkretnom programu, izuzetak nikada neće biti bačen, s obzirom da će se funkciji iz glavnog programa uvijek prenositi ispravni parametri (tj. parametri koji zaista imaju strukturu matrice). Međutim, s obzirom da dobro napisana funkcija ne treba ništa da zna o tome u kakvom će okruženju biti korištena, funkcija mora da se zna “odbraniti” u slučaju da u nju ipak “uđu” neispravni podaci.