

Zadaća 1.

Ova zadaća nosi ukupno 4 poena (od 20 poena, koliko će ukupno nositi sve zadaće zajedno), pri čemu svaki zadatak nosi po 1 poen. 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 petak, 29. III 2013. (do kraja dana) i ne može se produžiti. Zadaće se predaju putem Zamgera.

1. U teoriji brojeva susreće se pojam *palindromna otpornost*. Palindromna otpornost nekog nenegativnog cijelog broja definira se na sljedeći način. Broj se sabere sa brojem koji se dobija od cifara zadanog broja napisanog obrnutim redoslijedom (npr. za broj 35984 takav broj je 48953). Na novodobijeni zbir ponavlja se isti postupak, i tako daje, sve dok se kao rezultat ne dobije broj koji je jednak broju koji se dobija obrtanjem cifara. Broj koliko puta možemo obaviti taj postupak naziva se palindromska otpornost broja. Na primjer, za broj 35984 palindromska otpornost iznosi 6. Zaista, u skladu sa opisanim postupkom, imamo $35984 + 48953 = 84937$, $84937 + 73948 = 158885$, $158885 + 588851 = 747736$, $747736 + 637747 = 1385483$, $1385483 + 3845831 = 5231314$, $5231314 + 4131325 = 9362639$, a broj 9362639 je jednak broju koji se dobija obrtanjem cifara. U slučaju da je polazni broj već jednak broju koji se dobija obrtanjem njegovih cifara, njegova palindromska otpornost je 0. Za negativne brojeve, njigova palindromska otpornost jednaka je palindromskoj otpornosti njegove apsolutne vrijednosti (tj. broj -35984 također ima palindromsku otpornost 6).

Napišite funkciju nazvanu "PalindromskaOtpornost" koja kao parametar prima vektor cijelih brojeva a koja kao rezultat vraća novi vektor koji sadrži palindromske otpornosti odgovarajućih elemenata vektora koji je zadan kao parametar. Napisanu funkciju demonstrirajte u testnom programu koji traži da se sa tastature unese slijed brojeva u neki vektor, a koji nakon toga poziva napisanu funkciju da odredi palindromske otpornosti unesenih brojeva i ispiše ih na ekran (u vidu rečenica oblika "Palindromska otpornost broja x iznosi 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!

2. Za neku matricu A kažemo da ima *raspon* od (i_{min}, j_{min}) do (i_{max}, j_{max}) ako su ispunjeni sljedeći uvjeti:
 - a) Svi elementi matrice A_{ij} jednaki su nuli za $i < i_{min}$ i za $i > i_{max}$ (neovisno od toga kakav je j), pri čemu je makar jedan element matrice u i_{min} -tom redu i u i_{max} -tom redu različit od nule;
 - b) Svi elementi matrice A_{ij} jednaki su nuli za $j < j_{min}$ i za $j > j_{max}$ (neovisno od toga kakav je i), pri čemu je makar jedan element matrice u j_{min} -toj koloni i u j_{max} -toj koloni različit od nule.

Na primjer, sljedeća matrica ima raspon od (2, 2) do (3, 5):

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6 & 3 & 5 & 0 \\ 0 & 6 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Napišite funkciju "RasponMatrice" čija je svrha nalaženje opsega zadane matrice. Funkcija ima 5 parametara. Prvi parametar je matrica realnih brojeva organizirana kao vektor vektora. Funkcija treba da odredi raspon te matrice, i da smjesti nađene vrijednosti i_{min} , j_{min} , i_{max} i j_{max} u drugi, treći, četvrti i peti parametar respektivno. Ukoliko preneseni prvi parametar nema strukturu matrice (tj. ukoliko svi redovi proslijeđenog vektora vektora nemaju isti broj elemenata), funkcija treba baciti izuzetak koji se sastoji od teksta "Ulazni parametar nema formu matrice". Izuzetak također treba da se baci ukoliko su svi elementi u matrici nule. Naime, nije teško vidjeti da takva matrica nema raspona, tako da za nju treba baciti izuzetak koji se sastoji od teksta "Raspon nije definiran za nul-matricu".

Napisanu funkciju testirajte u glavnom programu na primjeru matrice čije se dimenzije i elementi unose sa tastature. Napomena: u ovakvom konkretnom programu, izuzetak "Ulazni parametar nema formu matrice" nikada neće biti bačen, s obzirom da će se funkciji iz glavnog programa

uvijek prenositi parametar koji zaista ima 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.

NAPOMENA: Vodite računa da su indeksi polja u koji se smještaju elementi matrice za 1 manji nego njihovi logički indeksi (tj. redni brojevi reda odnosno kolone u kojoj se element nalazi). Na primjer, element u trećem redu i drugoj koloni, čiji su logički indeksi 3 i 2, zapravo je smješten u polju sa indeksima 2 i 1 u odgovarajućoj strukturi podataka koja predstavlja matricu!

3. Potrebno je napisati funkciju koja određene riječi u nekom stringu zamjenjuje nekim drugim riječima. Prvi parametar je tipa “string” i on predstavlja string u kojem se vrši zamjena. Drugi parametar je vektor stringova (tj. vektor čiji su elementi tipa “string”) koji sadrži spisak riječi koje se mijenjaju (svaki element vektora sadrži po jednu riječ). Treći parametar je također vektor stringova, koji sadrži odgovarajuće riječi kojima se mijenjaju riječi iz prethodnog parametra. Drugi i treći parametar trebaju imati isti broj elemenata (u suprotnom, funkcija treba baciti izuzetak koji se sastoji od teksta “Nekorektni parametri”). Funkcija treba da kao rezultat vrati string koji se dobija nakon izvršenih zamjena. Na primjer, ukoliko string zadan prvim parametrom glasi “kako da ne”, te ukoliko se drugi i treći parametar sastoje respektivno od riječi “jabuka”, “da”, “kako”, “ne” i “majmun” odnosno “apple”, “yes”, “how”, “no” i “monkey”, kao rezultat funkcije treba da se dobije string “how yes no”.

Radi jednostavnosti, pretpostavite da string koji se transformira sadrži samo mala slova i razmake (i ništa drugo). Isto tako, pretpostavite da elementi vektora koji definiraju riječi koje se zamjenjuju kao i riječi sa kojima se zamjena vrši sadrže samo mala slova, bez razmaka. Ove pretpostavke ne morate provjeravati, već je samo uzmite kao takvu. Međutim, vodite računa da je moguće da riječi budu razdvojene sa više razmaka, kao i da se na početku odnosno kraju stringa mogu također nalaziti razmaci. Drugim riječima, string “ kako da ne ” uz pomoć istog “rječnika” kao i maloprije treba da se prevede u “ how yes no ”.

Treba paziti da se uvijek se zamjenjuju samo cijele riječi, nikad njihovi dijelovi. Na primjer, ukoliko string koji se transformira glasi “davor martić ima sestre maju i juliju” i ukoliko se vektori koji definiraju riječi koje se zamjenjuju odnosno riječi kojima se zamjena vrše sastoje respektivno od riječi “mart”, “maj” i “jul”, odnosno “ožujak”, “svibanj” i “srpanj”, kao rezultat transformacije string ostaje nepromijenjen, odnosno ne treba da postane “davor ožujakić ima sestre svibanju i srpanjiju”.

NAPOMENA: Nešto slično gore opisanom desilo se prilikom pokušaja automatskog “prevođenja” transkripta jedne parlamentarne sjednice sa bosanskog na hrvatski jezik uz pomoć Word-ove “Find & Replace” komande, kada je poslanik Martić postao Ožujakić. Interesantno je što bez obzira što Word ima opciju “Whole words only” kojom bi se izbjegao taj problem, “prevodioci” su je držali isključenom, jer su smatrali da na taj način “rješavaju” problem promjene riječi po padežima, koje inače “Find & Replace” ne može da “uhvati”.

4. Položaj zamišljenog robota koji može da se kreće kroz koordinatni sistem sa cjelobrojnim koordinatama opisuje se pomoću tri promjenljive “x”, “y” i “orjentacija”. Promjenljive “x” i “y” su tipa “int” i one čuvaju x odnosno y koordinatu pozicije na kojoj se robot trenutno nalazi. Promjenljiva “orjentacija” sadrži informaciju o pravcu u kojem robot trenutno gleda. Ona je tipa “Pravci”, koji predstavlja pobrojani tip definiran kao

```
enum Pravci {Sjever, Istok, Jug, Zapad};
```

Potrebno je napraviti program koji korisniku nudi sljedeće komande za upravljanje robotom: L – Nalijevo; D – Nadesno; I – Idi; K – Kraj. Komande L odnosno D obrću robota nalijevo odnosno nadesno za 90°. Komanda I obavezno je praćena jednim nenegativnim cijelim brojem (npr. I20) i ona pomjera robota za navedeni broj koraka u smjeru u kojem je robot trenutno usmjeren. Komanda K završava program. Sve druge komande su ilegalne, i trebaju dovesti do prijave greške i ponovnog izbora komande. Na početku rada, robot se nalazi na poziciji (0,0) i gleda na sjever. Dijalog između programa i korisnika trebao bi izgledati poput sljedećeg:

Robot se nalazi na poziciji (0,0) i gleda na sjever.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): D
 Robot se nalazi na poziciji (0,0) i gleda na istok.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I
 Pogrešna komanda (nedostaje parametar)!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I5
 Robot se nalazi na poziciji (5,0) i gleda na istok.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): D
 Robot se nalazi na poziciji (5,0) i gleda na jug.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I4
 Robot se nalazi na poziciji (5,-4) i gleda na jug.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): D
 Robot se nalazi na poziciji (5,-4) i gleda na zapad.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): S
 Pogrešna komanda!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): IXY2
 Pogrešna komanda (neispravan parametar)!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I2XY
 Pogrešna komanda (neispravan parametar)!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I0
 Robot se nalazi na poziciji (5,-4) i gleda na zapad.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I-1
 Pogrešna komanda (neispravan parametar)!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): I2
 Robot se nalazi na poziciji (3,-4) i gleda na zapad.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): L2
 Pogrešna komanda (suvišan parametar)!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): L
 Robot se nalazi na poziciji (3,-4) i gleda na jug.
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): KK
 Pogrešna komanda (suvišan parametar)!
 Unesi komandu (L - Nalijevo, D - Nadesno, I - Idi, K - Kraj): K
 Dovidjenja!

Budite sigurni 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.