

## Tekstovi zadatka

1. Napisati program koji definiše sledeće torke, ispisuje svaku torku na standardni izlaz, a zatim izračunava i ispisuje dužinu svake od kreiranih torki:
  - 1) torka koja sadrži celobrojne vrednosti 1, 2, 3 i 4
  - 2) torka koja sadrži jedan ceo broj 1, jedan string "dva", jedan decimalan broj 3.4 i jednu torku koja sadrži tri cela broja 5, 6 i 7
  - 3) jednoelementnu torku koja sadrži samo string "a"
  - 4) praznu torku
2. Napisati program koji definiše torku koja sadrži celobrojne vrednosti 1, 2, 3, 4 i 5 i ispisuje na standardni izlaz izdvojen prvi i poslednji element torke, a zatim pokušava da izmeni element na indeksu 2 postavljanjem na vrednost 0. Zakomentarisati zatim naredbu koja dovodi do greške u programu i definisati novu torku koja se od polazne razlikuje tako da na indeksu 2 sadrži vrednost 0. Novu torku dodeliti istoimenoj promenljivoj.
3. Napisati program koji definiše torku koja sadrži celobrojne vrednosti 1, 2, 3, 4 i 5 a zatim ispisuje na standardni izlaz tokre dobijene izdvajanjem odgovarajućih elemenata polazne torke:
  - 1) (2, 3, 4)
  - 2) (1, 3, 5)
  - 3) (4, 3, 2)
4. Napisati program koji definiše dve torke koje redom sadrže celobrojne vrednosti 1, 2, 3, 4 i 3, 4, 5, 6, 7, a zatim i treću torku koja se dobija nadovezivanjem prve dve torke. Program ispisuje torku dobijenu konkatencijom na standardni izlaz. Primetiti da se u dobijenoj torki brojevi 3 i 4 pojavljuju po dva puta.
5. Napisati program koji definiše torku koja sadrži celobrojne vrednosti 1, 2, 3, 4, 3, 5 i 3, a zatim proverava da li se u torki nalazi element sa vrednošću koja se učitava sa standardnog ulaza. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz, a u slučaju kada torka sadrži datu vrednost ispisuje i koliko puta se ta vrednost pojavljuje u torki kao i indeks prvog elementa sa tom vrednošću.
6. Napisati program koji definiše torku koja sadrži celobrojne vrednosti 1, 2, 3, 4 i 5 a zatim ispisuje njene elemente pojedinačno. Zadatak uraditi na tri načina:
  - 1) pomoću **while** petlje iteriranjem po indeksima elemenata
  - 2) pomoću **for** petlje iteriranjem po indeksima elemenata
  - 3) pomoću **for** petlje iteriranjem po elementima torke
7. Napisati program koji definiše torku koja sadrži stringove 'a', 'b', 'c', 'd', 'e' i 'f' a zatim pomoću kolekcijske **for** petlje i ugrađene funkcije **enumerate** ispisuje njene elemente sa pridruženim indeksima njihovih pozicija u torki.
8. Napisati program koji definiše dve torke, pri čemu prva torka sadrži stringove 'a', 'b', 'c' i 'd', dok druga torka sadrži decimalne vrednosti 1.1, 2.2, 3.3 i 4.4, a zatim pomoću ugrađene funkcije **zip** pravi novu torku koja sadrži uparene elemente ove dve torke. Program ispisuje torku dobijenu uparivanjem na standardni izlaz.
9. Napisati program koji definiše torku koja sadrži celobrojne vrednosti 9, -1, 5, 0 i -7 a zatim izračunava i ispisuje na standardni izlaz zbir elemenata, minimum i maksimum vrednosti u torki korišćenjem odgovarajućih ugrađenih funkcija za kolekcije.
10. Napisati program koji definiše sledeće skupove a zatim ispisuje tip njima pridruženih promenljivih kao i njihov sadržaj na standardni izlaz:
  - 1) skup koji sadrži celobrojne vrednosti 1, 2, 3 i 4
  - 2) skup koji sadrži celobrojne vrednosti 1, 2, 2, 3, 4, 4
  - 3) skup koji se dobija na osnovu liste [1, 2, 1, 3, 1, 4]
  - 4) prazan skup
11. Napisati program koji definiše skup koja sadrži stringove 'a', 'b', 'c', 'd', 'e' i 'f', izračunava i ispisuje broj elemenata skupa, a zatim pokušava da pristupi proizvoljnom elementu skupa pomoću operatora indeksiranja.

12. Napisati program koji definiše skup koja sadrži stringove 'a', 'b', 'c', 'd', 'e' i 'f' a zatim ispisuje na standardni izlaz elemente skupa pojedinačno.
13. Napisati program koji definiše skup koja sadrži celobrojne vrednosti 1, 2, 3 i 4 a zatim na dodaje i izbacije određene elemente i ispisuje sadržaj skupa na standardni izlaz nakon svake izmene, i to sledećim redosledom:
  - 1) skupu se dodaje element sa vrednošću 5
  - 2) skupu se dodaje element sa vrednošću 3 (duplikat)
  - 3) izbacuje se iz skupa element sa vrednošću 2
  - 4) izbacuje se iz skupa (nepostojeći) element sa vrednošću 6
14. Napisati program koji definiše dva skupa, pri čemu prvi skup sadrži brojeve 1, 2, 3 i 4, dok drugi skup sadrži brojeve 3, 4, 5, 6 i 7, a zatim izračunava uniju, presek, razliku i simetričnu razliku ova dva skupa i ispituje da li je prvi skup podskup ili nadskup drugog skupa. Svaku od navedenih skupovnih operacija izračunati na dva načina:
  - 1) pomoću odgovarajućih metoda klase **set**
  - 2) pomoću odgovarajućih skupovnih operatora
15. Napisati program koji definiše skup koji sadrži decimalne vrednosti 1.1, 2.2, 3.3, 4.4 i 5.5, a zatim proverava da li se u skupu nalazi element sa vrednošću koja se učitava sa standardnog ulaza. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz.
16. Napisati program koji definiše skup koji sadrži celobrojne vrednosti 9, -1, 5, 0 i -7 a zatim izračunava i ispisuje na standardni izlaz zbir elemenata, minimum i maksimum vrednosti u skupu korišćenjem odgovarajućih ugrađenih funkcija za kolekcije.
17. Napisati program koji definiše sledeće rečnike a zatim ispisuje tip njima pridruženih promenljivih kao i njihov sadržaj na standardni izlaz:
  - 1) rečnik koji sadrži sledeće parove ključ-vrednost: 'a' – 1, 'b' – 2 i 'c' – 3
  - 2) rečnik koji se dobija na osnovu liste parova, tj. torki [('a', 1), ('b', 2), ('c', 3), ('a', 1)]
  - 3) rečnik koji se dobija na osnovu skupa parova, tj. torki {'a', 1), ('b', 2), ('c', 3), ('b', 4)}
  - 4) prazan rečnik
18. Napisati program koji definiše rečnik koji sadrži sledeće parove ključ-vrednost: 'a' – 1.1, 'b' – 2.2 i 'c' – 3.3, a zatim proverava da li se u rečniku nalazi element sa ključem čija se vrednost učitava sa standardnog ulaza. Pristup elementima rečnika na osnovu ključa izvršiti na dva načina:
  - 1) pomoću operatora **[]**
  - 2) pomoću metoda **get**Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz.
19. Napisati program definiše inicijalno prazan rečnik u koji zatim dodaje sledeće parove ključ-vrednost: 1 – "jedan", 2 – "dva", 3 – "tri" i 2 – "DVA". Ispisati na standardni izlaz ovako dobijen rečnik.
20. Napisati program koji definiše rečnik koji sadrži sledeće parove ključ-vrednost: 'a' – 1, 'b' – 2, 'c' – 3 i 'd' – 4, a zatim uklanja iz rečnika element sa ključem 'b'. Ispisati na standardni izlaz sadržaj rečnika nakon uklanjanja elementa.
21. Napisati program koji definiše rečnik koji sadrži sledeće parove ključ-vrednost: 'a' – 1.1, 'b' – 2.2 i 'c' – 3.3, izračunava i ispisuje broj elemenata skupa, a zatim ispisuje na standardni izlaz sadržaj rečnika pomoću kolekcijske **for** petlje.
22. Napisati program koji definiše rečnik koji sadrži sledeće parove ključ-vrednost: 'a' – 1.1, 'b' – 2.2 i 'c' – 3.3 a zatim ispisuje: elemente pojedinačno. Zadatak uraditi na tri načina:
  - 1) sve ključeve u rečniku pojedinačno
  - 2) sve vrednosti u rečniku pojedinačno
  - 3) sve elemente rečnika pojedinačnoZa dobijanje liste svih ključeva, vrednosti i elemenata rečnika koristiti odgovarajuće metode.
23. Napisati program koji definiše rečnik koji sadrži sledeće parove ključ-vrednost: 'd' – 4.4, 'a' – 1.1, 'c' – 3.3 i 'b' – 2.2 a zatim na osnovu njega kreira nov rečnik koji je sortiran prema

ključevima i ispisuje sortirani rečnik na standardni izlaz. Zadatak uraditi uz pomoć ugrađene funkcije **sorted** primenjene na listu elemenata polaznog rečnika.