

Programiranje 1, praktični deo, 28.1.2017.

Na Desktop-u napraviti direktorijum sa imenom `P1_Januar1_2017_Tok_ImePrezime_BrojIndeksa` gde `tok`, `ime`, `prezime` i `broj indeksa` treba zameniti svojim podacima. Na primer, za studenta Jovana Marića sa trećeg toka čiji je broj indeksa 205/2015, ime direktorijuma je `P1_Januar1_2017_3_JovanMaric_ml15205`. Zadatke imenovati sa 1.c, 2.c, 3.c i 4.c i čuvati ih u napravljenom direktorijumu.

Primeri u zadacima koji su obeleženi zvezdicom su eliminacioni.

1. Napisati program koji sa standardnog ulaza učitava realan broj jednostruke tačnosti i ispisuje prvu cifru iza decimalnog zareza.

Primer 1*	Primer 2	Primer 3	Primer 4
<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: -2.546 IZLAZ: 5</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 12 IZLAZ: 0</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 234.834 IZLAZ: 8</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: -12.21 IZLAZ: 2</pre>

2. Napisati program koji učitava poene koje su studenti osvojili na ispitu (realne brojeve jednostruke tačnosti) do unosa broja -1 (ovo je oznaka za kraj učitavanja) i određuje koje su ocene dobili. Napisati funkciju `void odredi_ocene(float poeni[], int n, unsigned ocene[])` koja na osnovu niza sa poenima dužine `n` pravi niz ocena na sledeći način: za broj poena iz intervala `[0, 51)` student dobija ocenu 5, `[51, 61)` ocenu 6, `[61, 71)` ocenu 7, `[71, 81)` ocenu 8, `[81, 91)` ocenu 9, `[91, 100]` ocenu 10. Ipak, ukoliko nekome nedostaje manje od jednog poena da bi položio ispit, pustiti ga da položi sa ocenom 6. Testirati rad funkcije iz glavnog programa i ispisati dobijene ocene. U slučaju greške ispisati -1 na standardni izlaz za grešku. Pretpostaviti da neće biti više od 100 brojeva.

Primer 1*	Primer 2	Primer 3	Primer 4
<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 55 89.2 100 -1 IZLAZ: 6 9 10</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 50.2 78 77 -1 IZLAZ: 6 8 8</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 99 81 50.9 23 -1 IZLAZ: 10 9 6 5</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: -1 IZLAZ: -1</pre>

3. Napisati program za rad sa vektorima u trodimenzionom prostoru.

- a) Definisati strukturu koja predstavlja vektor iz trodimenzionog prostora (vektor s tri realne koordinate).
- b) Napisati funkciju `vektor vektorski_proizvod(vektor* u, vektor* v)` koja računa vektorski proizvod vektora na čije adrese pokazuju `u` i `v`.
- c) Napisati funkciju `double skalarni_proizvod(vektor* u, vektor* v)` koja računa skalarni proizvod vektora na čije adrese pokazuju `u` i `v`.
- d) Napisati program koji proverava da li su tri vektora koji se učitavaju sa standardnog ulaza komplanarni. Ukoliko su vektori komplanarni, ispisati 1 na standardni izlaz, a ukoliko nisu, ispisati 0.

UPUTSTVO:

Tri vektora su komplanarna akko je zapremina paralelepipeda koji oni razapinju 0 (tj. akko je jedan linearna kombinacija preostala dva).

Vektorski proizvod računa se po sledećoj formuli: $(u_1, u_2, u_3) \times (v_1, v_2, v_3) = (u_2v_3 - u_3v_2, u_3v_1 - u_1v_3, u_1v_2 - u_2v_1)$.

Skalarni proizvod računa se po sledećoj formuli: $(u_1, u_2, u_3) \cdot (v_1, v_2, v_3) = u_1v_1 + u_2v_2 + u_3v_3$.

Zapremina paralelepipeda razapetog vektorima `u`, `v`, `w` data je sa: $u \cdot (v \times w)$

Umesto poređenja realnog broja sa nulom, proveravati da li je apsolutna vrednost tog broja manja od 10^{-5} .

Primer 1*	Primer 2	Primer 3
<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 2.34 5.56 8.76 0 0.31 0 76.1 7 -1 IZLAZ: 0</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 1 2 -1 2 -1 1 8 1 1 IZLAZ: 1</pre>	<pre>INTERAKCIJA SA PROGRAMOM: ULAZ: 1 2 3 7 11 -3.4 2.75 4.75 2.15 IZLAZ: 1</pre>

4. Napisati program koji šifrjuje sadržaj datoteke čije se ime i ključ za šifrovanje unose kao argumenti komandne linije. Ključ je nenegativan ceo broj ne veći od 128 koji određuje broj pozicija pomeranja karaktera u desno u okviru ASCII tabele. Sadržaj datoteke se šifrjuje tako što se na vrednost svakog karaktera u datoteci dodaje ključ i to ciklično, ukoliko se izade iz okvira ASCII tabele. Na primer, za ključ 53, šifrat karaktera A (čija je vrednost 65) je karakter v (čija je vrednost 118), dok je šifrat karaktera u (čija je vrednost 117) karakter * (čija je vrednost 42). Rezultat šifrovanja smestiti u datoteku `šifrovana.txt`. U slučaju greške ispisati -1 na standardni izlaz za grešku.

Primer 1*	Primer 2	Primer 2
<pre>Poziv: ./a.out podaci.txt 4 PODACI.TXT zimske carolije SIFROVANA.TXT ~mqwoi\$gevspmni</pre>	<pre>Poziv: ./a.out leksikon.txt 1 LEKSIKON.TXT Ana voli Milovana SIFROVANA.TXT Bob!wpmj!Njmpwbob</pre>	<pre>Poziv: ./a.out simpatija.txt -1 SIMPATIJA.TXT Milovan voli Anu IZLAZ: -1</pre>