

Ispit iz Programiranja 2

Trajanje: 150 minuta

Napomene:

- Pažljivo proučite Uputstvo pre popunjavanja Obrasca za odgovore.
- Vrednost odgovora: tačan = 5; netačan = -1.25; nevažeći (nula ili više zacrnjenih kružića) = 0.
- Na pitanjima se može osvojiti najviše 25 poena.
- Zadaci nose po 20 poena.

I. ZADACI

1) Neka se u datoteci **teniseri.txt** nalaze podaci o teniserima po sledećem formatu: šifra igrača (ceo broj), ime (najviše 30 znakova), prezime (najviše 30 znakova) i broj poena na ATP rang listi (ceo broj). U datoteci **wimbledon.txt** se nalaze podaci o plasmanu tenisera na ovom teniskom turniru. U svakom redu datoteke se nalazi šifra tenisera (ceo broj), broj poena koji brani na turniru (ceo broj) i broj poena koji je osvojio na turniru (ceo broj). Raspored tenisera u datotekama ne mora biti identičan, niti svi teniseri iz prve datoteke moraju postojati u drugoj datoteci. Napisati program na programskom jeziku C koji pročita sadržaj ulaznih tekst datoteka i formira jednostruko ulančanu listu, a zatim u izlaznu datoteku **atplist.txt** za svakog tenisera upiše novi broj poena po formatu kao u prvoj ulaznoj datoteci. Novi broj poena se dobija tako što se od starog broja poena na ATP listi oduzme broj poena koje teniser brani, a zatim doda broj poena koje je teniser osvojio na turniru. Voditi računa o ispravnom korišćenju zauzetih resursa.

2) Napisati program na programskom jeziku C koji vrši određenu obradu nad matricom celih brojeva maksimalnih dimenzija 20x20. Program treba da učitava dimenzije i elemente matrice, pronađe minimalni element u matrici, a zatim ispiše u posebnim redovima sve one vrste koje ne sadrže taj minimalni element u matrici. Program treba da ponavlja izvršavanje sve dok se sa standardnog ulaza ne učitaju nekorektna dimenzije matrice. Učitavanje, obradu i ispis matrice realizovati kao zasebne potprograme koji sa glavnim programom komuniciraju isključivo putem argumenata i povratnih vrednosti.

II. PITANJA

1) Koja od sledećih tvrdjenja su tačna na programskom jeziku C?

- Funkcija **fopen()** vraća vrednost **EOF** ukoliko ne može da otvori datoteku.
- Binarni operator **->** se koristi za pristup poljima strukture putem pokazivača.
- Veličina unije je uvek jednaka zbiru veličina svih komponentata unije.

2) Šta ispisuje sledeći program na programskom jeziku C?

<pre>#include <stdio.h> #include <string.h> int fja(char *s) { char i=0, *s1, *s2, c; s1 = s; s2 = s+strlen(s)-1; while(s1 < s2){ i = s2-s1; if (i%4) { *s1 = *--s2 + (*s1 < 'a' ? 'b' - 'A' : 'A' - 'a'); s1++; } else { c = *s1; *s1 = *s2; *s2 = c; } s1++, s2--; } return i; }</pre>	<pre>void main() { char s[8] = "ELEKFAK", i; i = fja(s); printf("%s%d", s, i); }</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------

A) aLkKFAK2

B) bLkKFAK1

(C) bLIKFAK2

3) Šta ispisuje sledeći program na programskom jeziku C ukoliko jednostruko ulančana lista sadrži redom brojeve 3 2 5 8 7? Smatrati da funkcija **ucitajlistu()** ispravno formira, a funkcija **pisilistu()** ispravno ispisuje sadržaj jednostruko ulančane liste.

<pre>typedef struct elem { int broj; struct elem *sled; } Elem; void obradi (Elem* lst) { Elem *lst1 = lst, *lst2 = lst->sled; if (!lst) return; while(lst2) { int t = lst1->broj; lst1->broj = lst1->broj > lst2->broj ? lst2->broj : t; lst2->broj = t > lst2->broj ? t : lst2->broj; if (!lst2->sled) break; lst1 = lst2->sled; lst2 = lst1->sled; } }</pre>	<pre>void main (){ Elem *lst = ucitajlistu(); obradi(lst); pisilistu(lst); }</pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------

(A) 2 3 5 8 7

B) 7 8 5 2 3

C) 2 3 8 5 7

4) Šta ispisuje sledeći program na programskom jeziku C ako se putem komandne linije proslede brojevi 3 4 7 5 3 5?

<pre>#include <stdlib.h> #include <stdio.h> void main (int argc, char *argv[]) { int i, b, a = atoi(argv[1]); do { b = atoi(argv[--argc]); if (b >= a) printf("%d ", a = b); } while (argc > 1); }</pre>

A) 5 7

(B) 5 5 7

C) 4 7

5) Šta ispisuje sledeći program na programskom jeziku C?

<pre>#include <stdio.h> int pp(int y, int *z, int a){ *z = y + a; a--; return *z > y ? pp(a, z, y) : a; }</pre>	<pre>void main(){ int x = 3, y = 1, z; z = pp(x,&y,y); printf("%d %d %d", x, y, z); }</pre>
----------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------

A) 3 2 0

(B) 3 2 -1

C) 3 1 0

6) Neka je **void concatenate(char *p, char *d)**; funkcija koja obavlja operaciju nadovezivanja stringa na koji ukazuje pokazivač **d** na string na koji ukazuje pokazivač **p**. Koja realizacija ove operacije je ispravna?

A)

while(*p++);

(B)

while(*p) p++;

C)

while(0!=*p) p++;

while(*(p++)!= *(d++)); while(*(p++) = *(d++)); while(*(d++) = *(p++));