

## Ispit iz Programiranja 2

Ispit traje 120 minuta

### Napomene:

- a) Pažljivo proučite Uputstvo pre popunjavanja Obrasca za odgovore.  
 b) Vrednost odgovora: tačan = **5**; netačan = **-1.25**; nevažeći (nula ili više zacrtnjenih kružića) = **0**.  
 c) Na pitanjima se može osvojiti najviše **15** poena. Prvi zadatak nosi **25** poena, dok drugi nosi **30** poena.

### I ZADACI



1) Napisati program na programskom jeziku C koji rešava jedan šahovski problem na tabli proizvoljnih dimenzija MxN. Šahovska tabla se predstavlja pomoću dve matrice karaktera (jedna za pozicije belih, a druga za pozicije crnih figura) i može sadržati sledeće karaktere: K – kralj, D – dama, T – top, L – lovac, S – skakač, P – pešak i O – prazno polje. Dimenzije table nisu poznate unapred, pa se mora vršiti dinamička alokacija memorije. Potrebno je odrediti i ispisati pozicije svih skakača zadate boje, koji u narednom potezu mogu da „pojedu“ neku drugu figuru, kao i oznake i pozicije figura koje mogu da pojedu. Skakač može da se pomera na jedno od polja koje je najbliže polju na kome se nalazi, a koje nije u istoj horizontali, vertikalni ili dijagonalni, kao na slici. Učitavanje table i obradu realizovati kao zasebne potprograme koji sa glavnim programom komuniciraju samo putem argumenata i povratnih vrednosti. Prilikom učitavanja, obezbediti da se u matricu učitavaju samo dozvoljeni karakteri. Ukoliko se unese pogrešan karakter, ponoviti unos. Voditi računa o ispravnom korišćenju dinamičke memorije.

2) Napisati program na programskom jeziku C koji razvrstava đake osnovnih škola u željene srednje škole. U svakom redu datoteke `skole.txt` se nalaze informacije o srednjoj školi u sledećem formatu: broj slobodnih mesta (ceo broj), naziv škole (niz znakova, koji može sadržati blanko znake, od najviše 30 karaktera). Svakoj školi se dodeljuje redni broj koji odgovara rednom broju linije teksta u kojem se škola nalazi. Redni brojevi škola počinju od 0. U datoteci ima podataka o najviše 400 škola. U binarnoj datoteci `zelje.pod` svaki zapis sadrži: prijavni broj đaka (ceo broj), ime i prezime đaka (dužine tačno 30 karaktera), prosečnu ocenu (realan broj) i redni broj srednje škole koju đak želi da upiše. Prijavni brojevi đaka su jedinstveni. Program treba da učitava podatke o školama i đacima, zatim da formira tekstualnu datoteku `upisi.txt` koja će za svakog đaka sadržati podatke o dodeljenoj srednjoj školi. Jedan red datoteke treba da ima sledeći format: prijavni broj đaka, me i prezime đaka, prosečna ocena, naziv škole. Ako nema mesta u školi koju je đak naveo kao želju, umesto upisane škole upisati **NEUPISAN**. Đaci sa višom prosečnom ocenom imaju veći prioritet pri raspoređivanju. Prednost prilikom upisa đaka sa istim prosekom imaju oni sa nižim prijavnim brojem. Voditi računa o ispravnoj upotrebi resursa.

### II PITANJA

1) Šta ispisuje sledeći program napisan na programskom jeziku C?

```
#include <stdio.h>
#define ADD(x, y) x+y
#define MUL(x, y) x*y
void main() {
    int arr[6] = { 012, -2, 7 - 011, 0x1F, 3 }, res = 0;
    int n = sizeof(arr) / sizeof(int), i, j;
    for (i = -1, j = n; i <= n / 2;) {
        switch (--j, ++i) {
            case 0: res += MUL(arr[i], arr[j]);
            case 1: res += ADD(arr[i], arr[j]); break;
            case 4: res += MUL(ADD(arr[i], arr[j]), ADD(arr[i + 1], res));
            default: res++; break;
        }
    }
    printf("%d", res);
}
```

(A) 30

B) 29

C) 27

2) Šta ispisuje sledeći program napisan na programskom jeziku C?

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
enum e {N1=5, N2, N3, N4};
void main() {
    char *a, *b, *c, i, j;
    a = (int*)calloc(N2, sizeof(char));
    b = c = (int*)malloc(N4+1);
    strcpy(b, "31131402");
    while (*c) a[*c++ - '0']++;
    for (c = a; *c; printf("%c", *c++ + 'a')) ;
    free(a); free(b);
}
```

(A) bdbcb

B) abcde

C) acaba

3) Šta ispisuje sledeći program napisan na programskom jeziku C? Program je pokrenut sledećom komandom:  
`./program.exe polisemija metonimija 0x1 2`

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void f(char *a, char *b, int c) {
    if (a < b && c>0) f(a++, b--, c - 1);
    *a = *a < *b ? *a : ++*b;
}
void main(int argc, char *argv[]) {
    int c = atoi(argv[argc - 1]);
    char *p = argv[atoi(argv[3]) + 1],
        *q = p + strlen(p) - 1;
    f(p, q, c);
    printf("%s", p);
}
```

(A) blisemilb

B) betonimijb

C) netonimila

4) Šta treba da stoji umesto ##### i \$\$\$\$ da bi funkcija `reverse` ispravno obrtala dvostruko ulančanu nekrugu listu brojeva? Funkcija kao argumente prima dvostruke pokazivače na početak i kraj liste.

```
typedef struct elem {
    struct elem *next; struct elem *prev;
    int value;
} Elem;
void reverse(Elem** head, Elem** tail) {
    Elem *temp1, *temp2, *p = *head;
    while (p != NULL) {
        temp1 = p;
        temp2 = p->next;
        #####
    }
    $$$$
}
```

(A)

```
####:
p->next = p->prev;
p = p->prev = temp2;
$$$$:
*tail = *head;
*head = temp1;
```

B)

```
####:
p->prev = temp2;
p->next = temp1->prev;
p = temp2;
$$$$:
*tail = *head;
*head = temp1;
```

C)

```
####:
p->next = temp1->prev;
p = p->prev = temp2;
$$$$:
*tail = *head;
*head = temp2;
```