

Strukture podataka i algoritmi 1
Test - 13 poena

10.4.2021.

Ime i prezime	Broj indeksa	Broj poena
---------------	--------------	------------

1. Napisati rezultate sledećih kodova (2 poena, svaki primer po 0.5 poena):

a)

```
int a = 19;
int *p = &a;
&p = a;
*p = 20;
printf("%d %d", a, *p);
```

Greška pri kompajliranju.

c)

```
#define IZRAZ(x) (x + x * x)

int main() {
    int x = 6;
    printf("%d", IZRAZ(x + 4));
}
```

44

b)

```
int *x;
int p = 12;
x = &p;
p = 32;
p = *x;
printf("P = %d\n", p);
```

P = 32

d)

```
putchar(putchar('a'));
```

aa

2. Objasniti funkciju realloc i dati primer korišćenja. (1 poen)

Funkcija realloc služi za realokaciju memorije koja je prethodno dinamički alocirana. Za njeno korišćenje potrebno je uključiti biblioteku stdlib.h.

```
int *a = NULL;
a = realloc(a, 10 * sizeof(int));
```

3. Šta je rezultat sledećeg koda? (1 poen)

```
#include <stdio.h>
```

```
int avg(int *niz, int n) {
    int *avg = niz;
    int i;
    for (i = 0; i < n; i++)
        avg += niz[i];
    avg = niz;
    return *avg;
}
```

2 4

```
int main() {
    int niz[] = { 10, 20, 30, 40 };
    int n = sizeof(niz) / sizeof(int);
    printf("%d %d ", avg(niz, n) / n, n);
}
```

4. Šta je rezultat sledećeg koda? (1 poen)

```
#include <stdio.h>

void f1(int *y) {
    static int x = 2;
    x--;
    *y = (x) ? ((x--) ? --x : ++x) : x++;
    x += 1;
}

int f(int *y) {
    static int x = 2;
    f1(y);
}

int main() {
    int y;
    f(&y);
    printf("%d ", y);
    f(&y);
    printf("%d\n", y);

    return 0;
}
```

-1 -3

5. Šta je rezultat sledećeg koda? (1 poen)

```
#include <stdio.h>

int main() {
    int num1 = 4, num2 = 3;

    num1 ^= num2;
    num2 ^= num1;
    num1 ^= num2;

    printf("%d\n", num1);
    printf("%d\n", num2);

    return 0;
}
```

3

4

6. Napisati program koji računa sumu prethodno unetog niza. Program implementirati bez korišćenja operatora [] prilikom unosa članova niza i dodavanja članova u sumu. (1.5 poen)

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int i, n, sum = 0;
    int *niz;

    scanf("%d", &n);
    niz = (int*) malloc(n * sizeof(int));

    for(i = 0; i < n; i++) {
        scanf("%d", niz + i);
        sum += *(niz + i);
    }

    printf("Sum = %d", sum);
    return 0;
}
```

7. Napisati funkciju **void brojKrajnjihNula(unsigned n)** koja treba da odredi broj krajnjih nula u binarnoj reprezentaciji broja x. Funkcija treba da ispiše sa koliko nula se završava broj. (2 poena)

```
void brojKrajnjihNula(unsigned n) {
    int count = 0;

    while(n != 0){
        if(n & 1 == 1) break;
        n = n >> 1;
        count++;
    }
    printf(" %d \n", count);
}
```

8. Napisati program kojim se štampaju svi trocifreni brojevi koji su jednaki sumi faktoriijela svojih cifara. Faktoriyel broja odrediti preko rekurzivne funkcije. (2 poena)

```
#include <stdio.h>

long factorial(int n) {
    if (n == 1)
        return 1;
    else
        return n * factorial(n - 1);
}

int main() {
    int i;
    int sum, j, d, s;
    for (i = 100; i < 1000; i++) {
        sum = factorial(i % 10) + factorial((i / 10) % 10) + factorial(i % 100);

        if (i == sum)
            printf("%d", i);
    }
}
```

9. Napisati rekurzivnu funkciju **int DigitSum(int num)** koja pronalazi zbir cifara prirodnog broja. (1.5 poena)

```
int DigitSum(int num) {
    if(n1 == 0)
        return 0;
    return ((n1 % 10) + DigitSum(n1 / 10));
}
```