

Strukture podataka i algoritmi 1
I test – max 13 poena

2.4.2022.

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;
    p=&a;
    a=12;
    a=*p;
    printf("%d %d", a, *p);
```

12 12

c)

```
#define PROSEK(x, y, z) ((x+y+z) / 3)

int main() {
    int x = 6;
    int y = 10;
    int z = 3;
    printf("%d", PROSEK(x + 4, y - 3, z * 2));
}
```

7

```
int* pc, c;
c = 22;
pc = &c;
c = 11;
int *a;
a=pc;
*pc = 2;
*a=-9;
printf("C: %d\n", c);
```

C: -9

```
d) int x = 6;
    int y = 10;
    int z = 3;
    int *p=x+y+z;
    printf("P: %d\n", *p);
```

Kompajlerska greška

2. a) Objasniti šta rade komande **ls** i **cd** na Linux operativnom sistemu.(0.5 poena)
b) Šta predstavlja pokazivač? Dati primer definicije i postavljanja vrednosti. (0.5 poena)

a) **ls** - izlistava sadržaj trenutnog radnog direktorijuma

cd - vrši promenu trenutnog radnog direktorijuma

b) Pokazivač je promenljiva koja sadrži adresu druge promenljive

```
int *p;
```

```
*p=7;;
```

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

```

#include <stdio.h>
int fun(int n1)
{
    if(n1<10)
    {
        return 1;
    }
    else
    {
        return 1+fun(n1/10);
    }
}
int main()
{
    int x=19780123;
    int y=547811;
    int z=fun(x)*fun(y);
    printf("%d",fun(z));
}

```

2

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

```

#include<stdio.h>
main(){
    int c=6,x=5; float r = 2.5;
    if (c++ + ++c - c-- - --c) {
        printf("%f\n",(float)c/x);
    } else {
        printf("%f\n",(float)(c/x)); printf("%d\n",(int)(c/r));
    }
}

```

1.000000

2

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

```

#include <stdio.h>

```

```

#define CONST 11
int function(int x,int y) {
    static int tmp = CONST;
    (y>x)?((y>tmp)?(tmp=x):(tmp=tmp)):(tmp=y);
    return tmp;
}

main() {
    int a=5,b=15;
    printf("%d\n",function(a,b));
    a= 13;
    b= CONST/2;
    printf("%d\n",function(a,b));
}
5
5

```

6. Napisati funkciju koja broji koliko bitova je potrebno promeniti u jednom celom broju kako bi se dobio drugi ceo broj. Napisati program koji učitava 2 cela broja, i ispisuje koliki broj bitova je potrebno promeniti u prvom broju, kako bi bio dobijen drugi broj. Npr. **16 (0 0 0 1 0 0 0 0)** i **7 (0 0 0 0 0 1 1 1)** => potrebno je promeniti **4 bita**. **Napomena:** neophodno je koristiti bitovske operacije. (2.5 poena)

```

#include <stdio.h>
#include <string.h>

int countBits(int num1, int num2)
{
    int cnt = 0;
    int lsb1 = 0;
    int lsb2 = 0;

    while ((num1 > 0) || (num2 > 0)) {
        lsb1 = num1 & 1;
        lsb2 = num2 & 1;

        if (lsb1 != lsb2)
            cnt++;

        num1 = num1 >> 1;
        num2 = num2 >> 1;
    }
    return cnt;
}

```

```

}

int main()
{
    int num1 = 0;
    int num2 = 0;

    printf("Enter number1: ");
    scanf("%d", &num1);

    printf("Enter number2: ");
    scanf("%d", &num2);

    printf("Number of bits flipped: %d\n", countBits(num1, num2));
    return 0;
}

```

7. Napisati program kojim se štampaju svi trocifreni brojevi čija je suma cifara jednaka sumi svih jedinica u binarnom zapisu svake od cifara. (2.5 poena)

Primer:

116 (suma cifara - **1** + **1** + **6** = 8), binarno: **1** - 1, **1** - 1, **6** - 110 (1 + 1 + 1 + 1 + 0 = 4) - ne štampa se.

101 (suma cifara - **1** + **0** + **1** = 2), binarno: **1** - 1, **0** - 0, **1** - 1 (1 + 1 = 2) - štampa se.

```

#include <stdio.h>

int count1s(unsigned int num) {
    unsigned char i;
    int count=0;

    unsigned char totalBits=sizeof(num)*8;

    for(i=0;i< totalBits;i++) {
        if( num & (1<< i) )
            count++;
    }

    return count;
}

int main() {

```

```

    int bitsSum, sum;
    int i;
    int j, d, s;

    for (i = 100; i < 1000; i++) {
        j = i % 10;
        d = (i / 10) % 10;
        s = i / 100;

        bitsSum = count1s(j) + count1s(d) + count1s(s);
        sum = j + d + s;

        if (bitsSum == sum) {
            printf("%d\n", i);
        }
    }

    return 0;
}

```

8. Napisati rekurzivnu funkciju koja niz celih brojeva koji se sastoji **samo** od 0 i 1 pretvara iz binarnog u dekadni broj. Npr. **a=[0,1,0,1,0,1,1,1] => 87** (2 poena)

```

#include <stdio.h>
#include <math.h>
int toBinary(int a[], int degree, int n)
{
    if(degree==0)
    {
        return a[n]*1;
    }
    else
    {
        return a[n]*pow(2,degree)+toBinary(a,degree-1,n+1);
    }
}
int main()
{

```

```
int a[]={0,1,0,1,0,1,1,1};  
int n=sizeof(a)/sizeof(int);  
printf("%d\n", toBinary(a,n-1,0));  
}
```