

1. Objasnite zašto sledeći program nije korektan.

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

```
#define Nmax 100
```

```
int a[Nmax], i;
```

```
int main()
```

```
{
    for (i = 0; i <= Nmax; i++) { a[i] = 0; }
    printf("OK\n");
    return 0;
}
```

RESENJE**Analizirajte adresni prostor promenljivih!!!**

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

```
#define Nmax 100
```

```
int a[Nmax], i;
```

```
int main()
```

```
{
    printf("Adresa promenljive i je %p\n", &i);
    printf("Adrese a[0], a[1], a[99], a[100] su redom %p %p %p %p\n", &a[0], &a[1], &a[99], &a[100]);
    for (i = 0; i <= 100; i++) { a[i] = 0; }
    printf("OK\n");
    return 0;
}
```

2. Napisati C funkciju koja ce XORovati dva cela broja bez upotrebe bitovskog operatora ^.

ULAZ IZLAZ

1 2 3

3 5 6

RESENJE 01

```
int funkcijaXOR(int x, int y)
{
    return (x | y) & (~x | ~y);
}
```

RESENJE 02

```
int funkcijaXOR(int x, int y)
{
    int res = 0; // rezultat

    // 32-bit int
    for (int i = 31; i >= 0; i--)
    {
        // tekuci bitovi za x i y
        short b1 = x & (1 << i);
```

```

short b2 = y & (1 << i);

// ako su oba 1 onda 0, a inace xor je isti kao OR
bool xoredBit = (b1 & b2) ? 0 : (b1 | b2);

// novi rezultat
res <<= 1;
res |= xoredBit;
}
return res;
}

```

3) Koji od sledeća tri segmenta koda ne sadrži greške i pritom ispisuje vrednost 1? Smatrati da su pri prevođenju uključena sva potrebna standardna zaglavlja.

<pre> void f(){ int a[10]; int* b = a; void* k = b; b++; k = b; k++; printf("%d", k-b); } </pre>	<pre> void g(){ char a[10]; char *b = malloc(10); printf("%d", sizeof(a) == sizeof(b)); } </pre>	<pre> void h(){ char s1[] = "Drogram", s2[] = "Program"; s1[0]='P'; printf("%d", s1==s2); } </pre>
--	--	--

4. Šta ispisuje sledeći program napisan na programskom jeziku C?

```

#include<stdio.h>
int func(const char *s1, const char *s2);
void main () {
char *s1 = "Vrapci", *s2 = "zrna";
printf("%d", func(s1, s2));
}

```

```

int func(const char *s1, const char *s2) {
char *sc1 = s1, *sc2;
while (*sc1) { sc2 = s2;
while (*sc2 && *sc2 != *sc1) sc2++;
if (*sc2 == *sc1) return sc1 - s1;
sc1++;
}
return -1;
}

```

5. Šta ispisuje dati program, čija je izvršna verzija u datoteci program.exe, ukoliko se pokreće preko komandne linije sledećom komandom: program.exe 5 3 1 7 ?

```

#include <stdio.h>
#include <stdlib.h>
void main(int argc, char *argv[]) {
int x, y, j, s = 0;
x = atoi(argv[1]);

for (j = argc - 1; j > 1; j--) {
y = atoi(argv[j]);
s += x > y ? x : y;
}
}

```

```
printf("%d %d %d", x, y, s);
}
```

6.

Šta ispisuje sledeći program na programskom jeziku C, preveden kao sep2013.exe? Program je pozvan sledećom komandom: sep2013 3 8 2 8.

```
#include <stdio.h>
#include <stdlib.h>
int f(int* a,int b) {
    if ((*a)-- >= b-4 && b > 0){
        int c = b; return c + f(&b, *a); }
    return *a + b;
}
int main(int argc, char* argv[]) {
    int a = atoi(argv[3]);
    int b = atoi(argv[2]);
    printf("%d", f(&b,a));
    return 0;
}
```

7. Šta ispisuje sledeći program na programskom jeziku C?

```
#include <stdio.h>
#include <stdlib.h>
void f1(int *p){
    (*p++)--;
}
void f2(int *p){
    *p += *(p+1);
}
void main() {
    void (*f[])(int *) = {f1, f2, f2, f1};
    int niz[6] = {3, 4, 5, 6, 7, 8};
    int *p = niz, j;
    for (j = 0;j < 2;j++){
        (*f[*p % 4])(p);
        p = niz + *p % 5;
    }
    printf("%d",*p);
}
```

8. Šta ispisuje sledeći program na programskom jeziku C, preveden kao Jul2013.exe? Program je pozvan sledećom komandom: Jul2013 HIPOTEZA HIPOTENUZA HIPNOZA.

```
#include <stdio.h>
void main(int argc, char* argv[]) {
    char *s1 = *++argv, *s2;
    while (*s1++) {
        s2 = argv[1]++;
        while (*s2++) {
            if (*s1 == *s2) {
                printf("%c",*s1);
                break;
            }
        }
    }
}
```

9. Šta ispisuje sledeći program na programskom jeziku C, ako se redom unose vrednosti 5 7 8 6 2?

```
#include <stdio.h>
#include <stdlib.h>
void f(int **p){
    if (!(*p))
        *p = malloc(sizeof(int));
    scanf("%d", *p);
}

int main() {
    int *p[2] = {0}, i = 0, j;
    for (j = 0; j < 5; j++){
        f(p + i);
        i = (*p[i] + 1) % 2;
    }
    printf("%d %d", p[0][0], p[1][0]);
    free(p[0]); free(p[1]);
}
```