

---

### 1) Trivijalni program

```
#include <stdio.h>

int main()
{
    printf("Strukture podataka\n");
    return 0;
}
```

---

### 2) Zbrajanje

```
#include <stdio.h>

int main()
{
    int prvi, drugi;

    printf("Unesi prvi broj: ");
    scanf("%d", &prvi);
    printf("Unesi drugi broj: ");
    scanf("%d", &drugi);
    printf("%d + %d = %d\n", prvi, drugi, prvi + drugi);
    return 0;
}
```

---

### 3) Kvadratna jednadzba

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

int main()
{
    float a, b, c;
    float D;
    float x1, x2;

    printf("Unesi a: ");
    scanf("%f", &a);
    printf("Unesi b: ");
    scanf("%f", &b);
    printf("Unesi c: ");
    scanf("%f", &c);

    D = b * b - 4 * a * c;
    if(D >= 0)
    {
        x1 = (-b - sqrt(D)) / (2 * a);
        x2 = (-b + sqrt(D)) / (2 * a);
        printf("x1 = %f\n", x1);
        printf("x2 = %f\n", x2);
    }
    else
        printf("Rjesenja su kompleksni brojevi!");

    return 0;
}
```

---

#### 4) Pronalazenje najveceg elementa u (nesortiranom) polju

```
#include <stdio.h>

int main()
{
    int a[10], i;

    for(i = 0; i < 10; i++)
    {
        printf("Unesi %d. broj: ", i + 1);
        scanf("%d", &a[i]);
    }

    printf("\n");

    int max = a[0];

    for(i = 1; i < 10; i++)
        if (a[i] > max)
            max = a[i];

    printf("Najveci uneseni broj je %d\n", max);

    return 0;
}
```

---

#### 5) Sortiranje (s mjerenjem vremena)

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

void InsertionSort(int a[], int n);

int main()
{
    int a[10] = {4, 7, 3, 9, 0, 1, 6, 8, 2, 5};
    double timedif;
    double time1 = (double)clock();
    time1 = time1/(double)CLOCKS_PER_SEC;

    InsertionSort(a, 10);

    for(int i = 0; i < 10; i++)
        printf("%d ", a[i]);
    printf("\n");

    timedif = (((double)clock()) / (double)CLOCKS_PER_SEC) - time1;
    printf("Program je radio %f sekundi\n", timedif);
    return 0;
}

void InsertionSort(int a[], int n)
{
    for(int j = 2; j < n; j++)
    {
        int k = a[j];
        int i = j - 1;
        while(a[i] > k && i >= 0)
        {
            a[i + 1] = a[i];
            i--;
        }
        a[i + 1] = k;
    }
}
```

---

## 6) Pogadanje brojeva

Funkcija `rand()` vraća slučajan broj između 0 i `RAND_MAX`. (Obično je `RAND_MAX` jednak 32767).

Ako želimo dobiti slučajan prirodan broj između `m` i `n` ( $m < n$ ) uključujući i granice `m` i `n`, tada treba napraviti ovo:

```
int x = m + (n - m + 1) * ((float)rand() / (RAND_MAX + 1));

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

int main()
{
    int br;
    int pogodak = 0;
    srand(time(NULL));
    int x = 1 + 100 * ((float)rand() / (RAND_MAX + 1));
    printf("Zamislilo sam broj između 1 i 100. Pogodi ga!\n");
    int brpok = 0;

    while(pogodak == 0)
    {
        printf("Unesi broj: ");
        scanf("%d", &br);
        brpok++;
        if(br == x)
        {
            printf("Bravo! Pogodio si u pokusaju br. %d\n", brpok);
            pogodak = 1;
        }
        else
        {
            if (x > br)
                printf("Trazeni broj je veci!\n");
            else
                printf("Trazeni broj je manji!\n!!");
        }
    }

    return 0;
}
-----
```

## 7) Funkcije ne mijenjaju svoje ulazne varijable!

```
#include <stdio.h>

void func(int);

int main()
{
    int a = 1;
    printf("Prije poziva funkcije func: a = %d\n", a);
    func(a);
    printf("Nakon poziva funkcije func: a = %d\n", a);
    return 0;
}

void func(int a)
{
    a = a + 100;
}
-----
```

## 8) Funkcije mijenjaju polja!

```
#include <stdio.h>

void func(int[]);

int main()
{
    int i;
    int a[3] = {0, 1, 2};
    printf("Prije poziva funkcije func:\n");
    for(i = 0; i < 3; i++)
        printf("%d ", a[i]);
    printf("\n\n");

    func(a);

    printf("Nakon poziva funkcije func:\n");
    for(i = 0; i < 3; i++)
        printf("%d ", a[i]);
    printf("\n");

    return 0;
}

void func(int a[])
{
    a[0] = a[0] + 100;
}

-----
```