

Εργαστήριο 8

Συναρτήσεις χωρίς παραμέτρους.
Παραμετρικές συναρτήσεις.
Τιμή επιστροφής συνάρτησης.
Αναδρομικές συναρτήσεις.

Παραδείγματα. Συναρτήσεις. Δηλώσεις ΚΑΙ ορισμοί.

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

void BadNews(){ printf(" You have a VIRUS \n"); }
// void, no parameters

void pri ( int n ){ printf("number is %d\n", n ); }
// void, one parameter

int myread(){ int n;

    printf("give number -> ");
    scanf("%d",&n);
    pri(n);

    return n; }

// returns int, no parameters, calls pri() so must be AFTER pri

int myfunc( int n ){ int a; a = 7*n+8; return a; }
// returns int, one parameter

float func2( int n, float x){ return x+n; }
// returns float, two parameters

main(){ int m;

    BadNews(); BadNews(); // call BadNews

    pri(3); // call pri with constant
    m=5; pri(m); // with variable
    pri( 4*m+8 ); // with expression

    m = myread(); // call myread, save result in m
```

```

m = myfunc(10); // call with constant, variable or expression
pri(m); // show result

printf("func2 = %f \n", func2(4,3.14) ); // shows 7.14

system("pause");
}

```

Παραδείγματα. Συναρτήσεις. Δηλώσεις, main, και μετά ορισμοί.

```

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

void BadNews();
int myread(); // myread before pri
void pri ( int n );
int myfunc( int n );
float func2( int n, float x);

main(){ int m;

    BadNews(); // call BadNews

    pri(3); // call pri

    m = myread(); // call myread, save result in m

    pri(myfunc(10)); // call myfunc inside pri

    printf("func2 = %f \n", func2(4,3.14) );

    system("pause");
}

// ορισμοί

void BadNews(){ printf(" You have a VIRUS \n"); }

int myread(){ int n;
    printf("give number -> ");
    scanf("%d", &n); pri(n);
    return n; }

```

```

// myread calls pri,
// but pri is already declared,
// so everything is OK

void pri ( int n ){ printf("number is %d\n", n ); }
int myfunc( int n ){ int a; a = 7*n+8; return a; }
float func2( int n, float x){ return x+n; }

```

Παραδείγματα. Αναδρομικές συναρτήσεις.

```

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

void crazy(){ crazy(); } // infinite recursion

main(){

    crazy();

    // calling crazy on Windows silently stops this program,
    // no warning!

    system("pause");
}

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

void test( int n ){

    printf("starting test with n = %d \n", n);

    if (n>0) test( n-1 ); // restricted recursion

    printf("ending test with n = %d \n", n);
}

main(){

    test(5); // fires 6 calls to test

    system("pause");
}

```

Παράδειγμα. Δυαδικό σύστημα.

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

void binary( int n )
{ if (n>0) { binary( n/2 ); printf("%d", n%2); } }

main(){

    binary(11); // shows number 11 in binary

    printf("\n\n"); // extra skip line

    system("pause");
}
```

Παραδείγματα. Άθροισμα ακολουθίας.

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

int sum( int n ){ if (n>0) return sum(n-1) + n; else return 0; }

main(){

    printf("Gauss sum = %d \n", sum(100) ); // 1+2+...+99+100 = 5050

    system("pause");
}
```