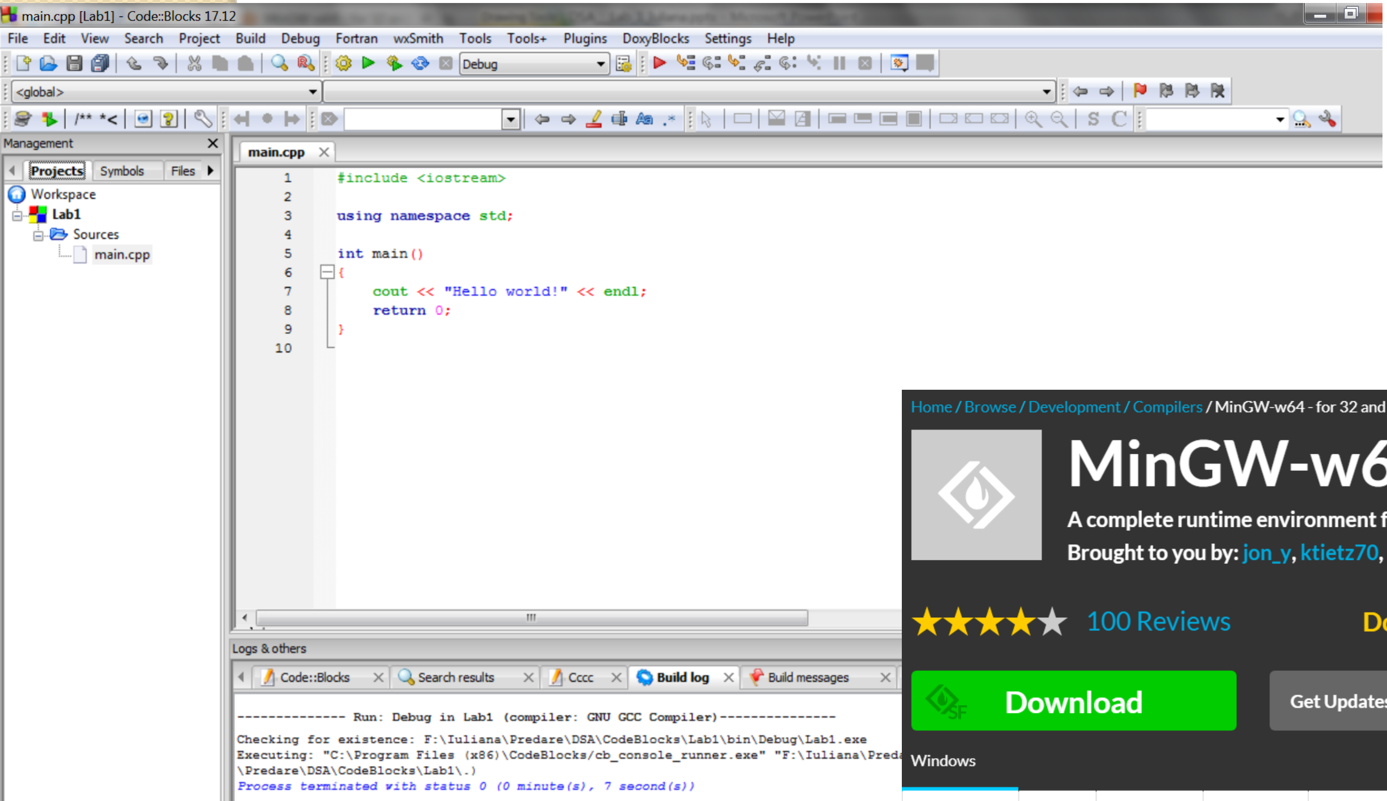
Data Structures and Algorithms – Lab 1

Iuliana Marin
marin.iulliana25@gmail.com
FILS

Tools

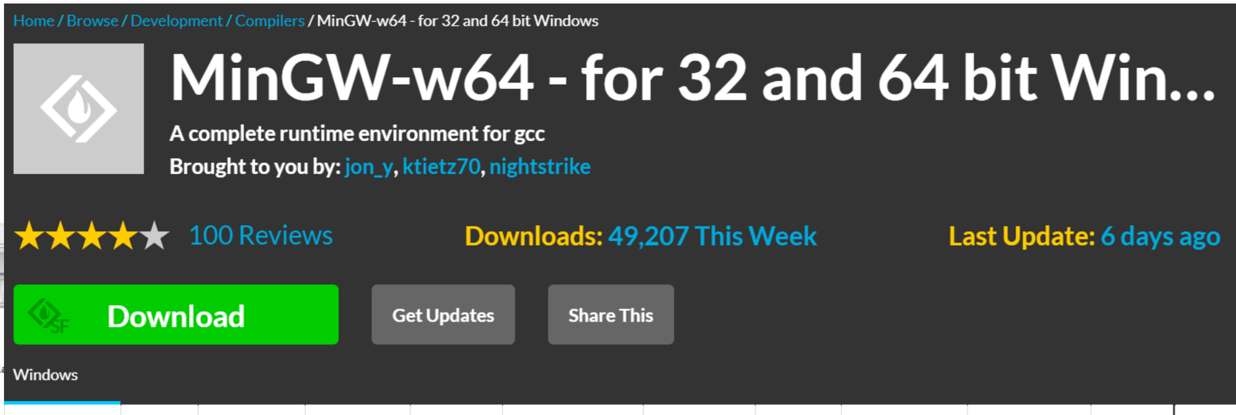
- **C-Free 4.0** (http://www.programarts.com/cfree_en/download.htm)

- Any other IDE or compiler for C/C++ (e.g. GCC under Linux)
  - Use CodeBlocks


main.cpp [Lab1] - Code::Blocks 17.12
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
Management
Projects Symbols Files
Workspace
Lab1
Sources
main.cpp
Logs & others
Code::Blocks
Search results
Cccc
Build log
Build messages

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!" << endl;
    return 0;
}
```

```
-------------- Run: Debug in Lab1 (compiler: GNU GCC Compiler)---------------
Checking for existence: F:\Iuliana\Predare\DSA\CodeBlocks\Lab1\bin\Debug\Lab1.exe
Executing: "C:\Program Files (x86)\CodeBlocks/cb_console_runner.exe" "F:\Iuliana\Preda
\Predare\DSA\CodeBlocks\Lab1\.)
Process terminated with status 0 (0 minute(s), 7 second(s))
```



Home / Browse / Development / Compilers / MinGW-w64 - for 32 and 64 bit Windows

MinGW-w64 - for 32 and 64 bit Win...

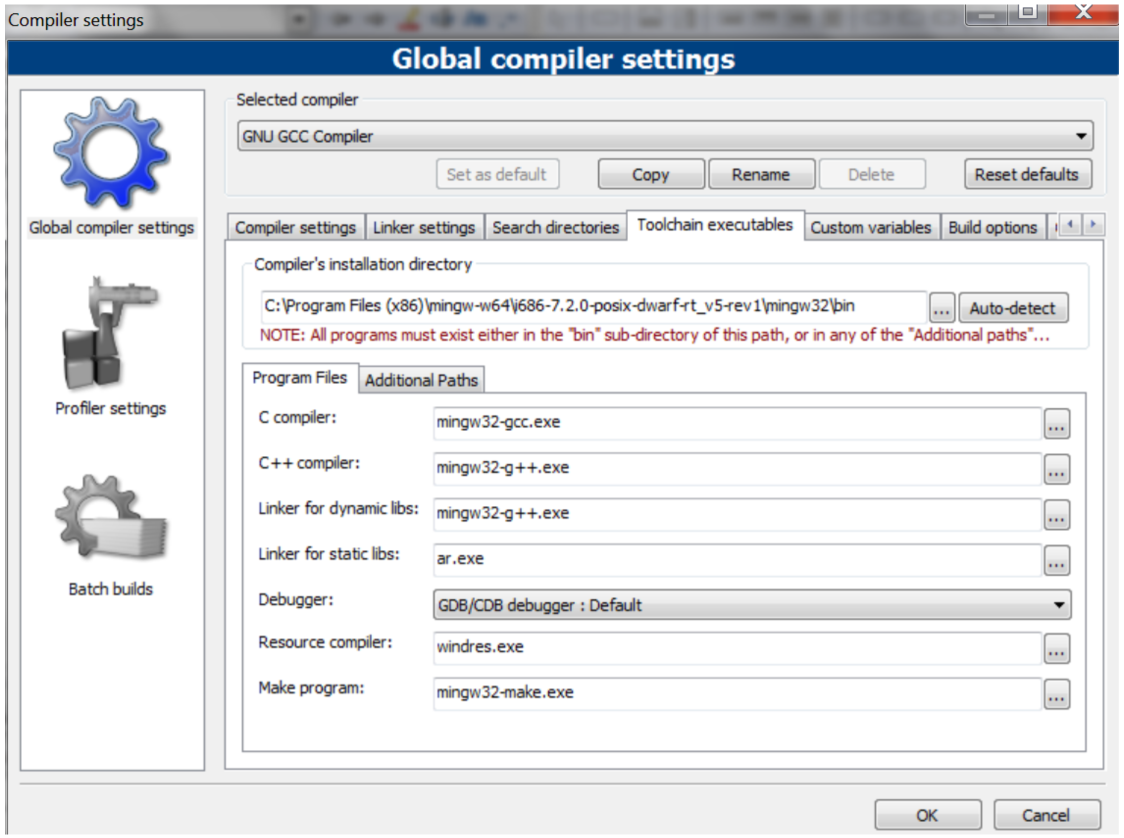
A complete runtime environment for gcc

Brought to you by: jon_y, ktietz70, nightstrike

100 Reviews Downloads: 49,207 This Week Last Update: 6 days ago

Set the compiler in CodeBlocks

- Go to Settings -> Compiler -> Toolchain executables and set the path towards your MinGW compiler.



Selected compiler: GNU GCC Compiler

Compiler's installation directory: C:\Program Files (x86)\mingw-w64\i686-7.2.0-posix-dwarf-rt_v5-rev1\mingw32\bin

NOTE: All programs must exist either in the "bin" sub-directory of this path, or in any of the "Additional paths"...

C compiler: mingw32-gcc.exe
C++ compiler: mingw32-g++.exe
Linker for dynamic libs: mingw32-g++.exe
Linker for static libs: ar.exe
Debugger: GDB/CDB debugger : Default
Resource compiler: windres.exe
Make program: mingw32-make.exe

Grading

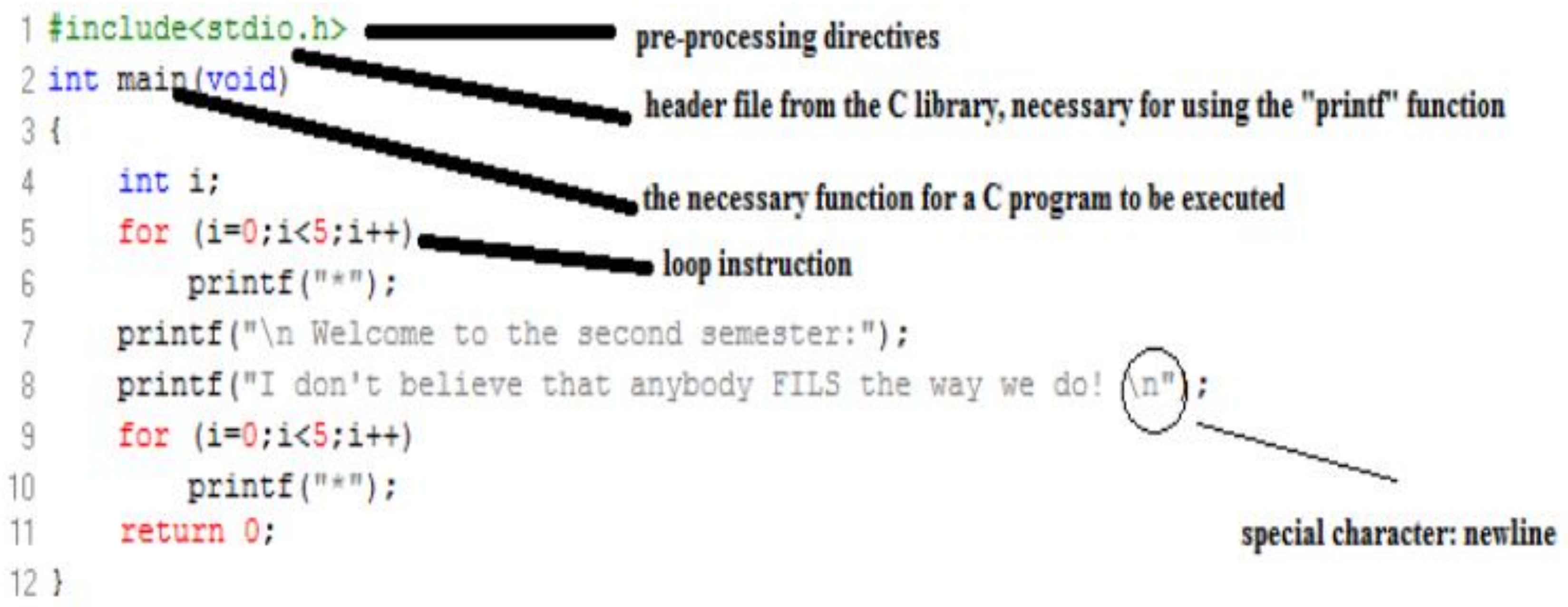
- Final grade = E1+E2+E3+E4+E5
- E1 – written exam during session: 45%
- E2 – 3 big homework in teams (2 students from the same group, same prof): 30% (10%+10%+10%)
- E3 – activity during labs (presence + class activity + small home assignments): 20%
- E4 – **unannounced** small tests during course: 5%
- E5 – bonus - max 2 pts. (for participation at scientific session, for correctly answering to qs during course, etc).
- Scientific session (participation 0.50 pts, participation and award received, 1 pt)
- Course qs (0.1-0,25 pts/qs)
- **Penalties (if you are asked smth during the course and you don't know)**
- Passing conditions: final grade>=4,5 (final grade can be >10) and E1 >=4.5
- Conditions to going to the exam: E2+E3+E4+E5 >= 2 and **presence at least 8 labs**

Objectives

- to run and compile C programs;
- to identify the structure of a C program;
- to use standard I/O operations;
- to define variables;
- to declare and implement functions;
- to make structures;

Exercise

Identify the structure of a typical C program! Attention, C is case sensitive!!!



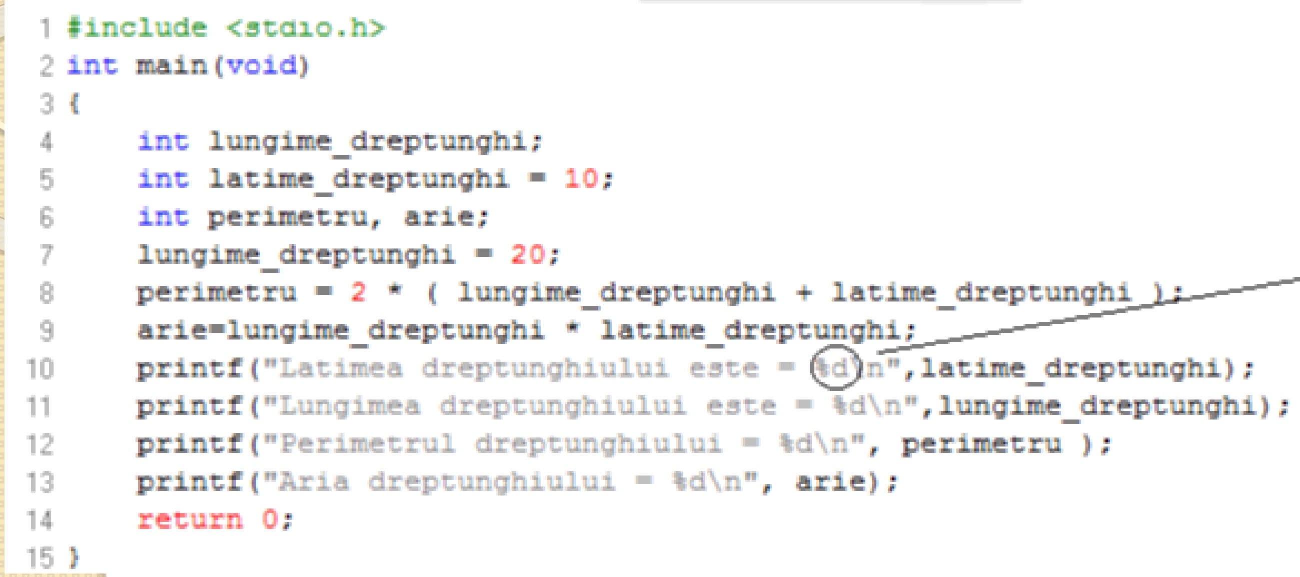
```
#include<stdio.h>
int main(void)
{
    int i;
    for (i=0;i<5;i++)
        printf("*");
    printf("\n Welcome to the second semester:");
    printf("I don't believe that anybody FILS the way we do! \n");
    for (i=0;i<5;i++)
        printf("*");
    return 0;
}
```

pre-processing directives
header file from the C library, necessary for using the "printf" function
the necessary function for a C program to be executed
loop instruction
special character: newline

A C program is written in a file with the “.c” extension: the source code.
After compilation, another file, with the “.o” extension appears: the object code.
After execution, another file, with the “.exe” extension appears: the executable.

To read the value of a from keyboard: scanf("%d", &a);
To print the value of a: printf("%d\n", a);
To finish the main function successfully: return 0;

Standard Output Operations



```c
#include <stdio.h>
int main(void)
{
     int lungime_dreptunghi;
     int latime_dreptunghi = 10;
     int perimetru, arie;
     lungime_dreptunghi = 20;
     perimetru = 2 * ( lungime_dreptunghi + latime_dreptunghi );
     arie=lungime_dreptunghi * latime_dreptunghi;
     printf("Latimea dreptunghiului este = %d\n",latime_dreptunghi);
     printf("Lungimea dreptunghiului este = %d\n",lungime_dreptunghi);
     printf("Perimetrul dreptunghiului = %d\n", perimetru );
     printf("Aria dreptunghiului = %d\n", arie);
     return 0;
}
```

format specifier for int variables

Other format specifiers:

%i or %d	int
%c	char
%f	float
%lf	double
%s	string

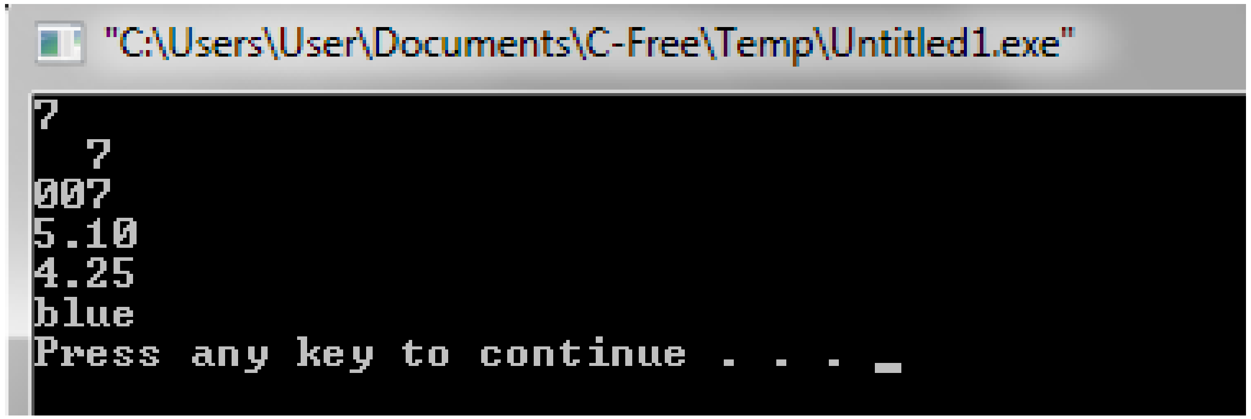
Signature of *printf* function

printf(control, par1, par2, …, parn);
Where
control = a string which defines the texts and the formats specifiers
par1, par2, …, parn = expressions; their values are written taking into account their type

Exercise

- Run the below example and see how each format specifier works

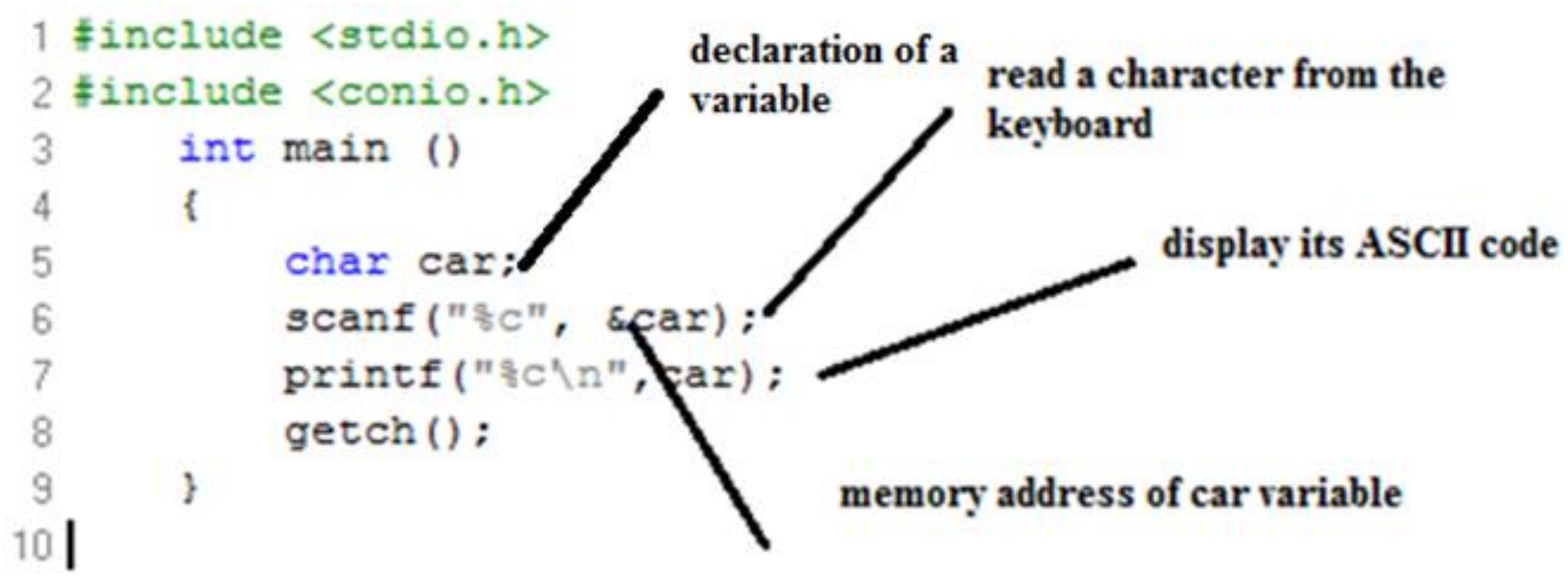
```
#include <stdio.h>
int main(void) {
        printf("%d\n", 7);
        printf("%3d\n", 7);
        printf("%03d\n", 7);
        printf("%3.2f\n", 5.1);
        printf("%.2f\n", 4.245);
        printf("%s\n", "blue");
        return 0;
}
```



"C:\Users\User\Documents\C-Free\Temp\Untitled1.exe"

```
7
  7
007
5.10
4.25
blue
Press any key to continue . . . _
```

Standard Input Operations



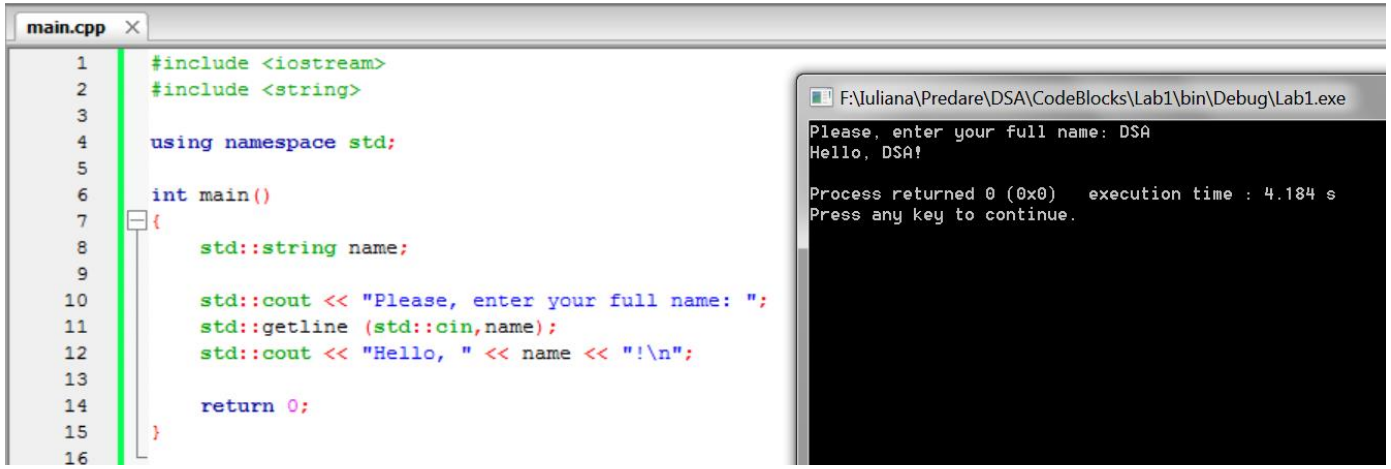
```
#include <stdio.h>
#include <conio.h>
      int main ()
      {
            char car;
            scanf("%c", &car);
            printf("%c\n", car);
            getch();
      }
```

declaration of a variable
read a character from the keyboard
display its ASCII code
memory address of car variable

Scanf has the same signature as *printf* and it is defined in *stdio.h*.

Using Strings

Include the string library by declaring the header *#include <string>*



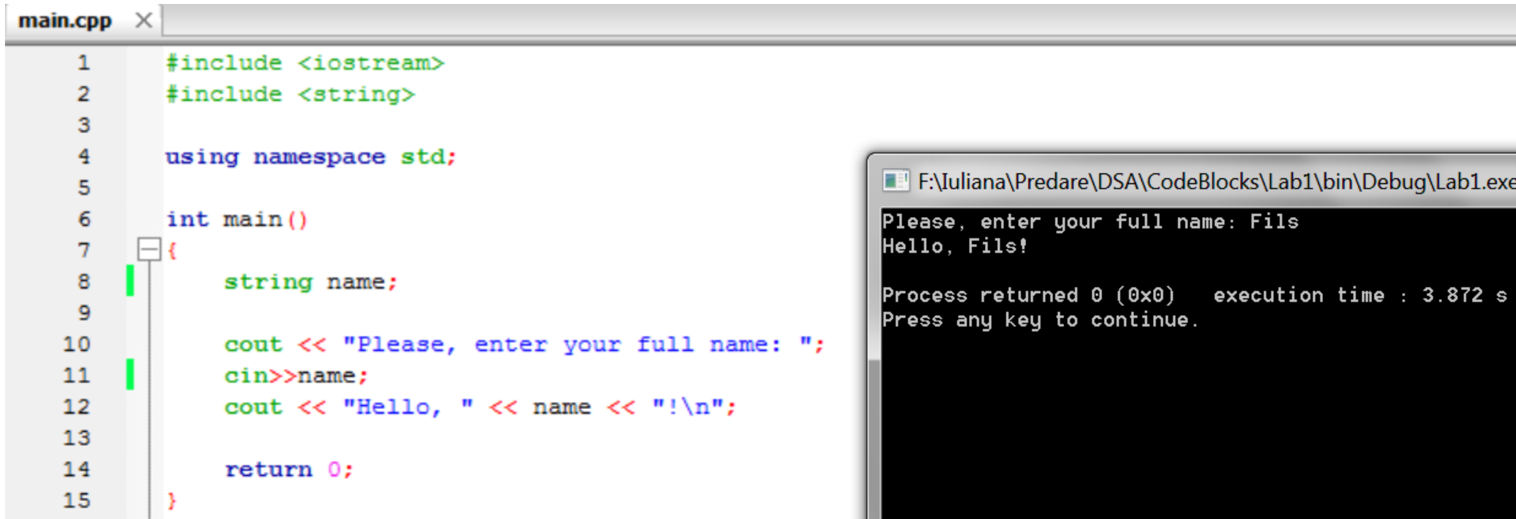
main.cpp

```cpp
#include <iostream>
#include <string>

using namespace std;

int main()
{
    std::string name;

    std::cout << "Please, enter your full name: ";
    std::getline (std::cin,name);
    std::cout << "Hello, " << name << "!\n";

    return 0;
}
```

F:\Iuliana\Predare\DSA\CodeBlocks\Lab1\bin\Debug\Lab1.exe

```
Please, enter your full name: DSA
Hello, DSA!

Process returned 0 (0x0)   execution time : 4.184 s
Press any key to continue.
```

OR without the std namespace in front



main.cpp

```cpp
#include <iostream>
#include <string>

using namespace std;

int main()
{
    string name;

    cout << "Please, enter your full name: ";
    cin>>name;
    cout << "Hello, " << name << "!\n";

    return 0;
}
```

F:\Iuliana\Predare\DSA\CodeBlocks\Lab1\bin\Debug\Lab1.exe

```
Please, enter your full name: Fils
Hello, Fils!

Process returned 0 (0x0)   execution time : 3.872 s
Press any key to continue.
```

Exercise

Write a program to calculate the average between two float numbers. The result shall be displayed with 2 decimals. Use *scanf* and *printf*!

%.2f -> format specifier for float with 2 decimals

Functions: declaration and implementation

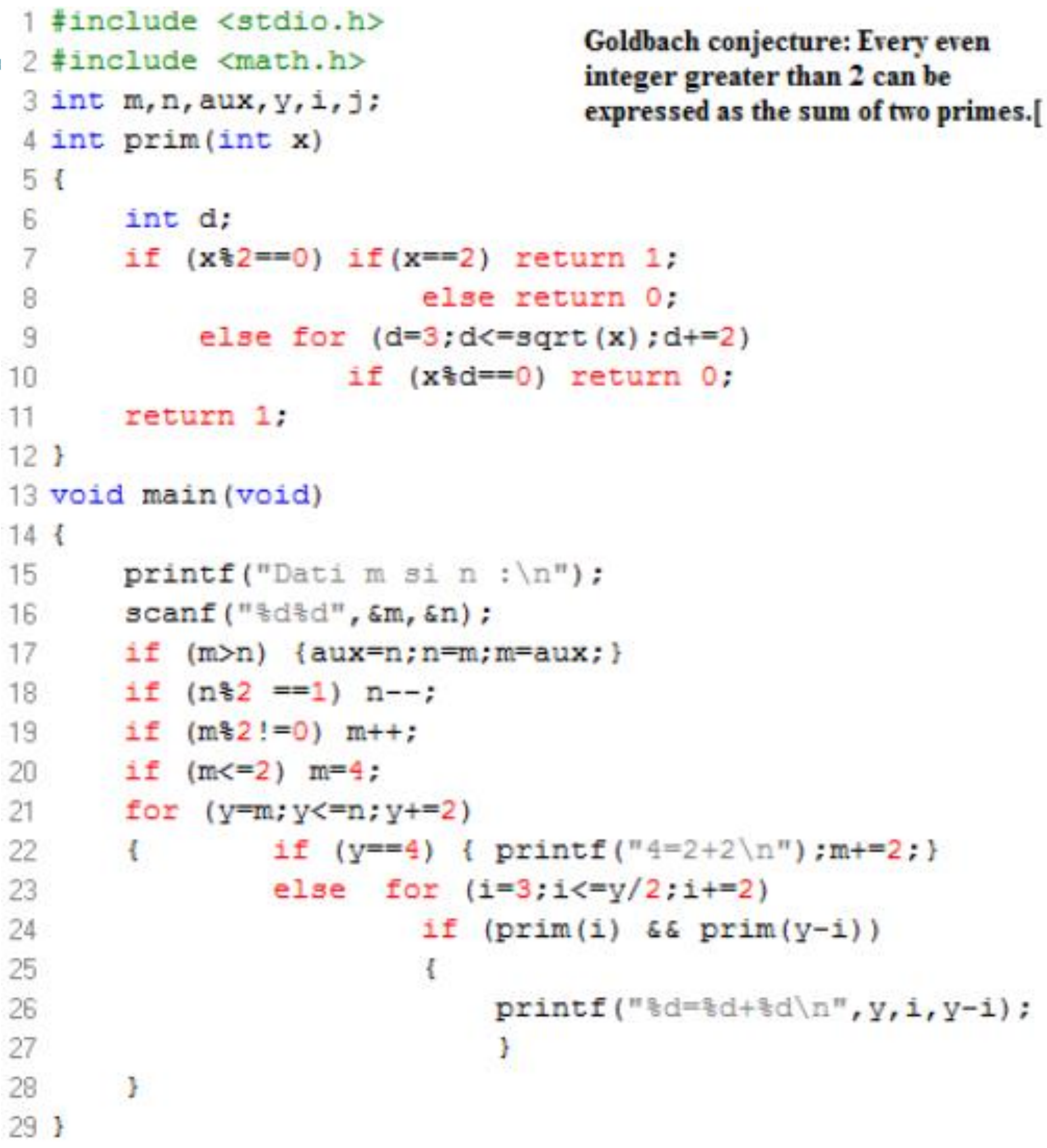
Signature:

```
type_of_the_returned_result function_name(list_of_formal_params)
{
declaration_of_local_variables;
instructions;
}
```

Visibility domain: local vs. global variables

Parameter passing: by-value

Example



```c
#include <stdio.h>
#include <math.h>
int m,n,aux,y,i,j;
int prim(int x)
{
    int d;
    if (x%2==0) if(x==2) return 1;
                else return 0;
        else for (d=3;d<=sqrt(x);d+=2)
                if (x%d==0) return 0;
    return 1;
}
void main(void)
{
    printf("Dati m si n :\n");
    scanf("%d%d",&m,&n);
    if (m>n) {aux=n;n=m;m=aux;}
    if (n%2 ==1) n--;
    if (m%2!=0) m++;
    if (m<=2) m=4;
    for (y=m;y<=n;y+=2)
    {       if (y==4) { printf("4=2+2\n");m+=2;}
            else  for (i=3;i<=y/2;i+=2)
                    if (prim(i) && prim(y-i))
                    {
                        printf("%d=%d+%d\n",y,i,y-i);
                        }
    }
}
```

Goldbach conjecture: Every even integer greater than 2 can be expressed as the sum of two primes.[

- Note the use *of math.h* library: for *sqrt* function (the same meaning as in Java)
- Note the control flow structures (if, if-else, for, …)
- Note the function definition and call: the implemented function calculates if a number is prime or not

Exercise

Check whether a number is a palindrome or not.

Hint: a palindrome is a number that remains the same when its digits are reversed.

333 is a palindrome
123 is not a palindrome

Structures

- a user-defined data type that allows grouping of heterogeneous elements;
- a collection of one or more variables (fields), grouped under one name;
- the members of a structure are accessed with “.”;

Example:

```
struct data {
		unsigned char day;
		unsigned char month;
		unsigned long year;
		char name_day[3];
		char name_month[4];
};

typedef struct data data;
data today; // data is now a type
```

```
typedef struct data {
unsigned char day;
unsigned char month;
unsigned long year;
char name_day[4];
char name_year[4];
} data;
data today;
```

typedef allows you to declare instances of a struct without using keyword "struct"

•signed char, which gives you *at least* the -127 to 127 range. (-128 to 127 is common)
•unsigned char, which gives you *at least* the 0 to 255 range.

Example: utilization

```
void writeDDMMMYYYY(data myDate)
{
        printf("%2d %s %4d ", myDate.day,
                myDate.name_month, myDate.year);
}
```

Exercise

Consider we have n computers for which we know the processor, the frequency of the processor in MHz, the RAM memory in MB and the capacity of the hard disk (in MB).

Display all the computers which can be connected to a network under an operating system with the frequency of at least x MHz.

The best computer having the maximum frequency and RAM will be chosen as a server.

HINT

```
#include <stdio.h>
typedef struct {
        int freq, ram, hd;
        char processor[256];
}computer;
computer c[25];
int i, n, x, y, z, server, bestfreq, bestram;

int main(void){…}
```

Homework

1. Design a structure for representing dates and write functions that:

- Check if a variable value of the structure is a valid date.
- Calculate the next date of a given date.
-Calculate the date before a given date.
-Check if the year is a leap one (contains 366 days).

2. Rare polynomials with integer coefficients are polynomials of large degrees and many coefficients equal to 0. They can be represented by a data structure defined as:

```
typedef struct
{
    int Coef;
    unsigned int Exponent;
} TMonom;
typedef TMonom TPolinom[50];
```

Write functions for writing, reading, addition and multiplication of rare polynomials .