

Introduction to Computer Programming with C Language

BY

Dr. MOHAMED ANWAR

MODIFIED BY

Dr. EMAD SAMI

C Language

Course Chapters

1. Introduction
2. Program development
3. The Essentials of C programs
4. Manipulating data with operators
5. Reading and writing standard I/O
6. Decision
7. Iteration
8. Arrays
9. C functions

1. Introduction

Chapter Objectives:

- 1-1 Computer Program
- 1-2 Programming Languages
- 1-3 The C programming Language
- 1-4 Interpreter
- 1-5 Compiler
- 1-6 Assembler
- 1-7 Setting up the computer
- 1-8 The C-Free 4.0 IDE Preparation Steps
- 1-9 The first C program

1-1 Computer Program

- A *computer program* is a sequence of instructions written to perform a specified task with a computer.
- The program has an executable form (i.e., after you write the program to perform a specific task and then save it, **.exe** file is obtained for running)
- The **.exe** program:
 - # It has human readable form called the source code
 - # this source code is written by computer programmers.
 - # this source code written using a programming language.
 - # an interpreter or compiler is used to convert the source code into **.exe** form.

1-2 Programming Languages

- It is an artificial language designed to communicate instructions to a computer.
- It has a text form (i.e., it contains words, numbers, and punctuation).
- Have three groups: *machine language*, *assembly language* and *high level language*.

1-2 Programming Languages (Cont's)

- machine language:
 - program code has 0's and 1's only.
 - difficult to read and write.
- assembly language:
 - Its level higher than machine language.
 - program code has mnemonic English codes. (ADD, SUM, SUB etc)
 - needs **assembler** to translate assembly code into machine code.
- high level language (HHL)
 - Its level higher than machine or assembly language.
 - program code has human language.
 - examples of HHLs are C, C++, VB, Java ... etc.
 - needs **compiler** or **interpreter** to translate HHL code into machine code.

1-2 Programming Languages (Cont's)

Advantages of high level language

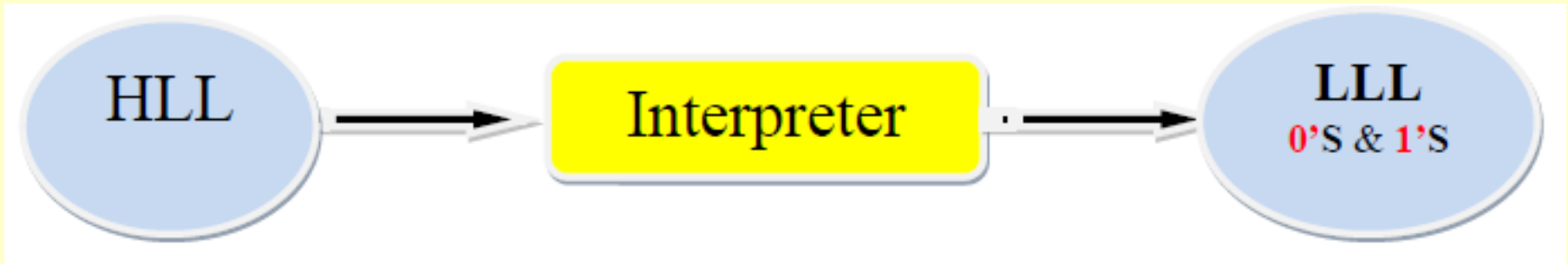
1. Readability: Programs are easy to read. سهل القراءة والفهم
2. Maintainability: Programs are easy to maintain. سهل حفظه وتذكره
3. Portability: Programs are easy to port across different computer platforms. سهل أن تقوم بتخزينه على هارد أو فلاشة مثلا
4. Reusability: Programs are saved into a library file and invoked them in next programming project simply by including the library file. يمكن (إستدعاء البرنامج الذى تم تخزينه) داخل (برنامج آخر يتم كتابته).

1-3 The C Programming Language

- * C is HHL.
- * C allows you to control computer hardware.
- * C sometimes called the lowest high level programming language.
- * Many other high level languages have been developed based on C.
- * American National Standards Institute (ANSI) defines the standard for C.
- * C language uses C Free Borland Compiler.

1-4 Interpreter

* is a program that converts High Level Language Program into Low Level Language (i.e., into binary code of 0's and 1's)

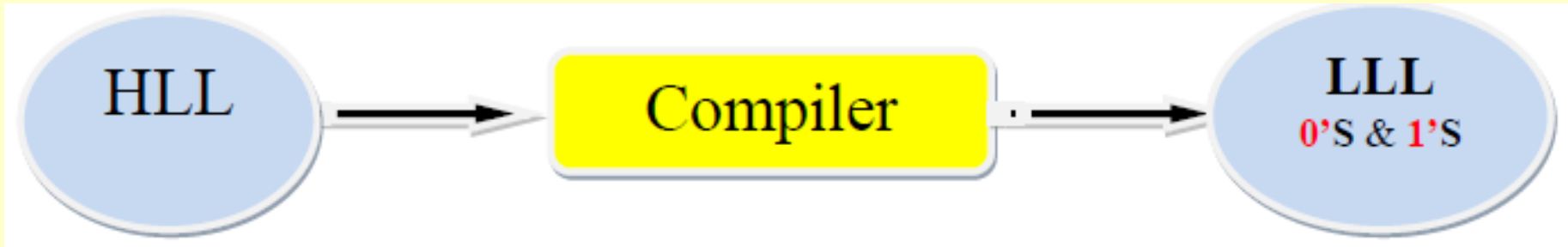


* “During program running; the **interpreter** converts each line in the C code into 0's and 1's”

* **Note:** *BASIC is interpreter based language*

1-5 Compiler

* is a program that converts High Level Language Program into Low Level Language (i.e., into binary code of **0**'s and **1**'s)



* “During program running; the **compiler** converts a whole block of the C code into 0’s and 1’s”

* **Note:** *C & C++ are compiler based languages*

1-6 Assembler

* is a computer program that takes computer instructions and converts them into a pattern of bits (bit is a program form of 0's and 1's)

1-7 Setting up the computer

* For using C Programming Language on you PC; your PC must have specifications not less than the following:

1- Processor: Pentium 4

2-HDD: 80Gb

3-RAM: 512 Mb

• C Programming Language that will use in this course:

C-Free 4.0 IDE

1-7 Setting up the computer (cont's)

* For writing a source code using C Programming Language:
you need:

1- Text Editor: to write the source code; but save it to extension “.C”

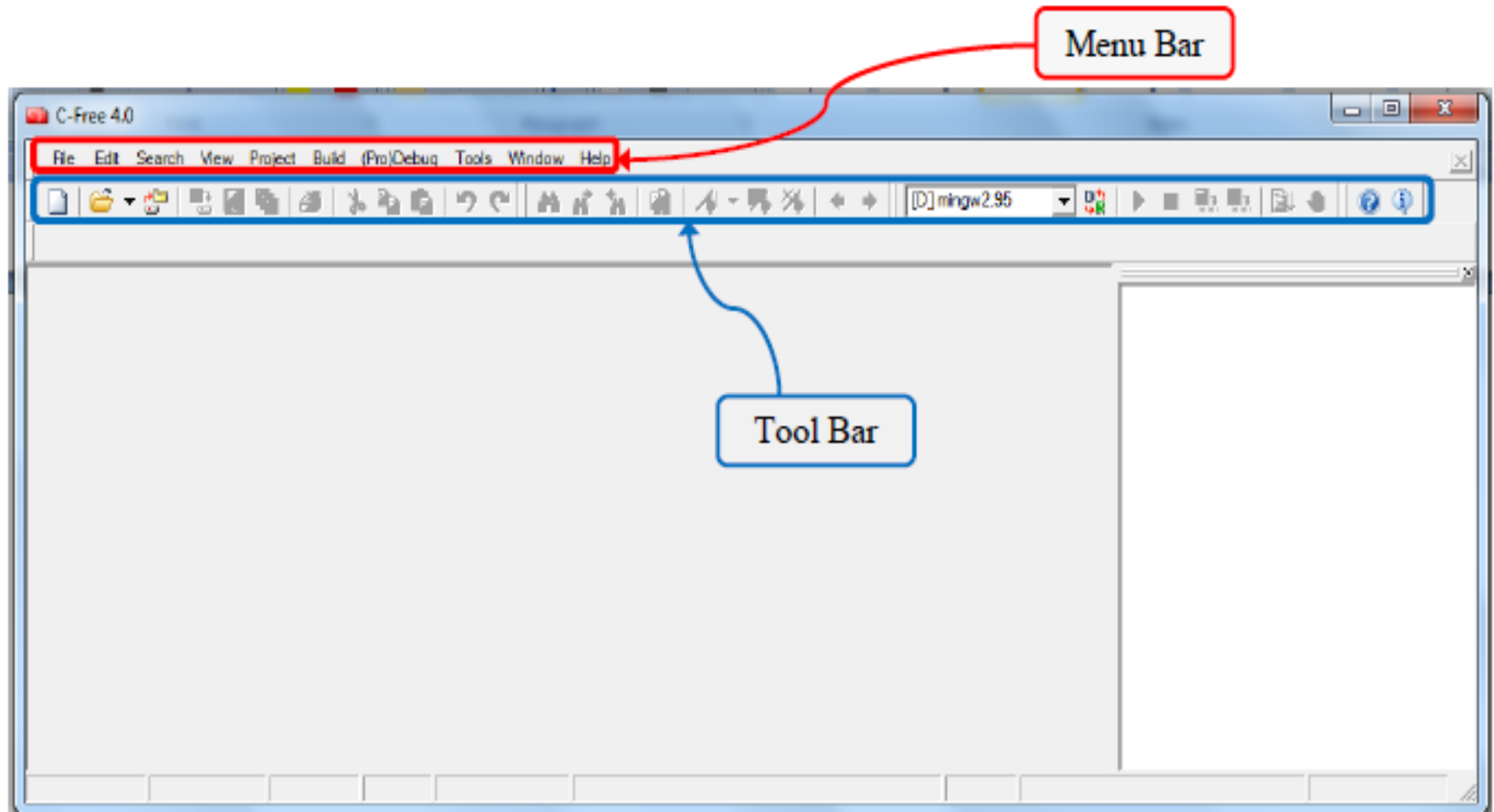
2- Compiler: to debug “i.e., finding and correcting errors” and convert the source code into executable file “.exe”, then run it.

Or

C-Free 4.0 software which is a professional C/C++ Integrated Development Environment (IDE) that: allow you to: edit, build, debug, and run programs freely.

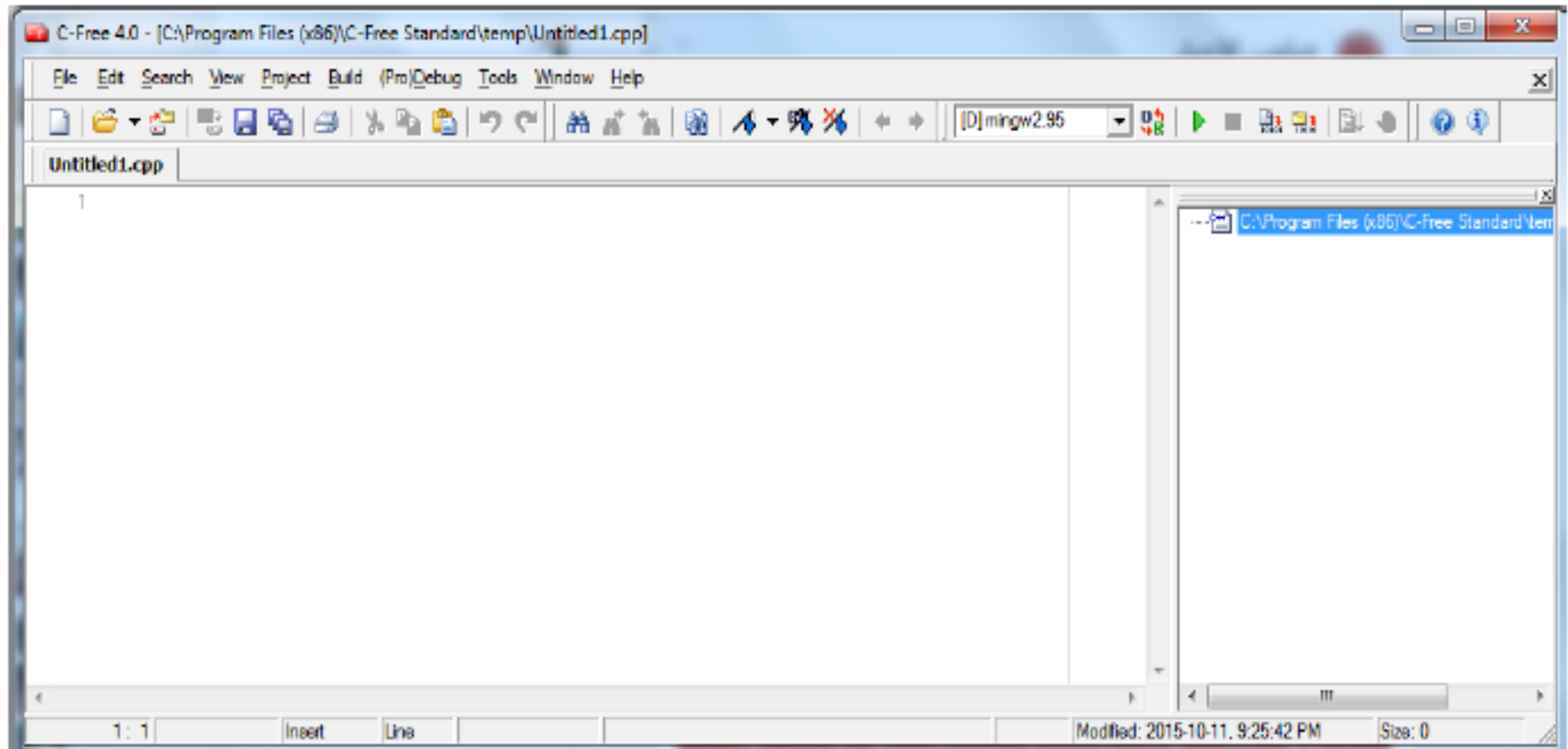
1-8 The **C-Free 4.0 IDE** Preparation Steps

- 1- Setup/Install C-Free 4.0 IDE on your HDD.
- 2- Run C-Free 4.0 IDE; you will see:



1-8 The C-Free 4.0 IDE Preparation Steps (cont's)

3- From Menu Bar, open File>New or Ctrl+N



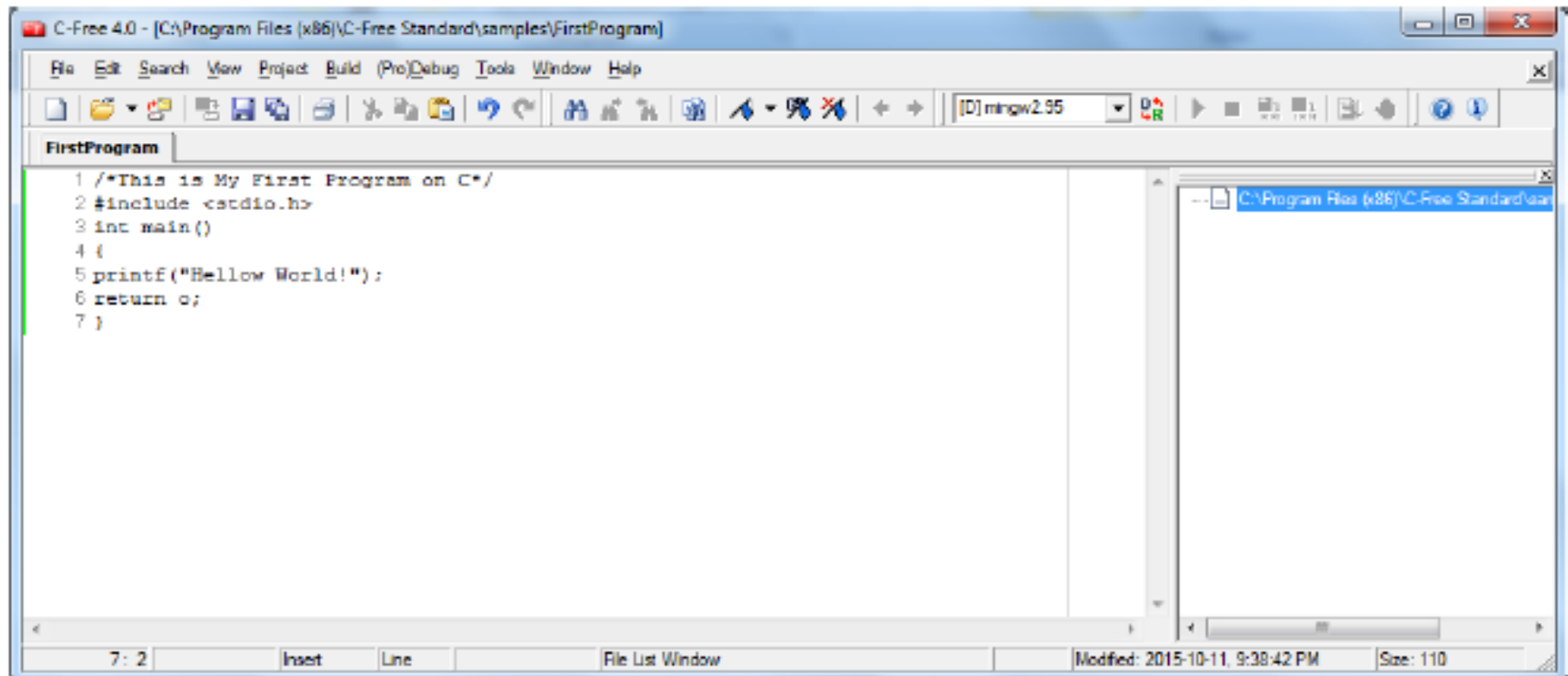
1-8 The C-Free 4.0 IDE Preparation Steps (cont's)

The First C Program

1- Open C Program by double click on the icon at the desktop:



2- **File>New** ... then edit the following program as shown

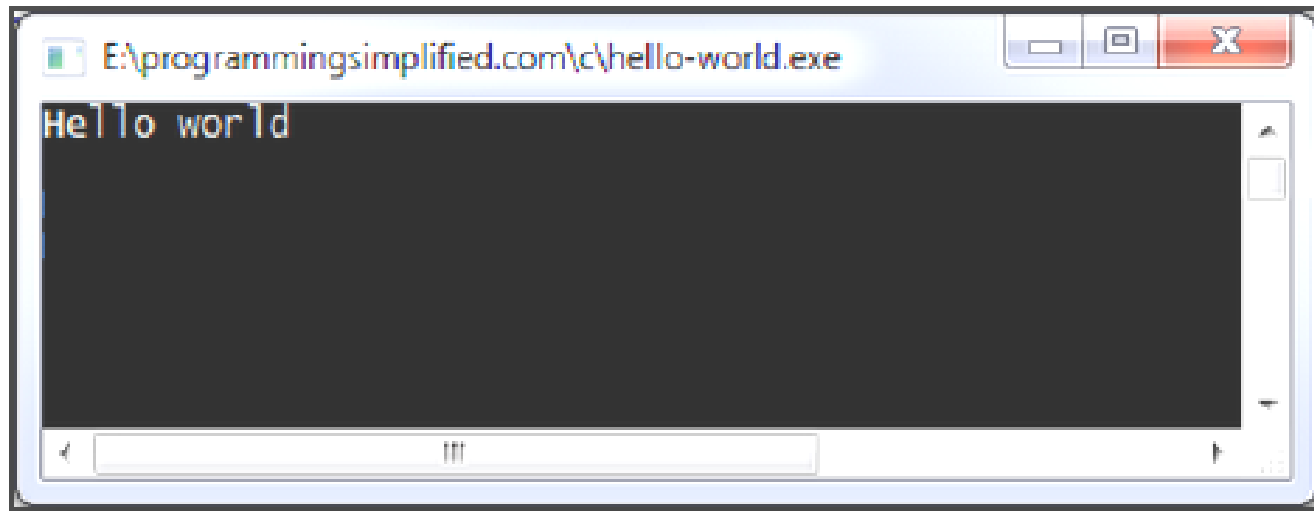
A screenshot of the C-Free 4.0 IDE. The window title is 'C-Free 4.0 - [C:\Program Files (x86)\C-Free Standard\samples\FirstProgram]'. The menu bar includes File, Edit, Search, View, Project, Build, (Pro)Debug, Tools, Window, and Help. The toolbar contains various icons for file operations and development. The main editor area shows a C program with the following code:

```
1 /*This is My First Program on C*/
2 #include <stdio.h>
3 int main()
4 {
5 printf("Hellow World!");
6 return 0;
7 }
```

The status bar at the bottom shows '7: 2', 'Insert', 'Line', 'File List Window', 'Modified: 2015-10-11, 9:38:42 PM', and 'Size: 110'.

1-8 The C-Free 4.0 IDE Preparation Steps (cont's)

- 3- **Debug or F9 ...** to check if errors or not in the program
- 4- **File>Save As ...** FirstProgram.c
- 5- **Run or F5 ...** run the program to appear the output



1-9 The first C program

- If You Want To Print “Hello World” On the Screen
- The source code is:

```
/* This is my First C Program */  
#include<stdio.h>  
int main ()  
{  
printf(" Hello World! \n ");  
return 0 ;  
}
```

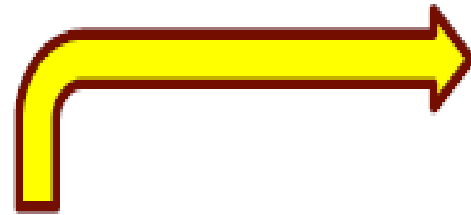
Comments

```
/* This is my First C Program */
```

- The compiler **ignore** everything between two **slashes** and **asterisks**.
- **C** compilers allows to write comments like

```
/*  
  
This comment does not increase the size of the executable file  
(Binary Code), nor does it affect the performance speed  
  
*/
```

Header Files



stdio.h Header File

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

- This **Header Files** required by **include** directive
- Header Files extensions **.h** means **header**
- The **stdio.h** header file contains numerous prototypes and macros to perform **input** or **output I/O** for C Program

Main Function

- Very **special** function
- Each C Program **MUST** have a main() function
- The main() function syntax as follows

```
int main ()
```

```
{
```

```
return 0 ;
```

```
}
```

Printing On Screen

- To print a text or numbers **printf** is used.
- Syntax **printf**(" ");
- The **new line** character **\n**
- To print **Hello World!** write the following statement

```
printf(" Hello World! \n ");
```

1. 7 The first C Program

Comments :

- Any line starts with a combination of slash and asterisk `/*`, and ends with `*/` is a comment.
- `/*` called the opening comment mark and `*/` is the closing comment mark.
- The C compiler ignores everything between the opening and closing comment mark.
- Adding comments into a C program does not increase the size of the binary code of the program
- Many C compilers now started with `//` to mark the beginning of a comment line and the comment ends at the end of the line.

1. 7 The first C Program

The #include directive :

- In C, #include forms a preprocessor directive that tells the C preprocessor to look for a file and place the contents of the file in the location where the #include directive indicates.
- The preprocessor is a program that does some preparations for the C compiler before your code is compiled.
- The #include directive asks the C preprocessor to look for and place stdio.h file where the directive is in the C program.

1. 7 The first C Program

The #include directive :

- The name of the `stdio.h` file stands for standards input output header file.
- The C programming language distinguishes between lowercase and uppercase characters. In other words, C is a case sensitive language.
- For instance, `stdio.h` and `STDIO.H` are different filenames in C.

1.7 The first C Program

Header files :

- The files that are required by the #include directive, like `stdio.h`, are called header files because #include directives are almost always placed at the head of C programs.
- Actually, the extension name of `.h` means “header.”
- Several header files will be found during this book such as `string.h`, `math.h`, and so on.

1.7 The first C Program

Angle brackets (< >) and double quotes (“ “):

- Angle brackets (< >) ask the C preprocessor to look for a header file in a directory other than the current one.
- If we use double quotes (“ “), the C preprocessor looks in the current directory first before it goes elsewhere for the stdio.h header file.

1. 7 The first C Program

The main Function :

- In line 3 of the program, you see this function:
 main ()
- This is a very special function in C.
- Every C program must have a main () function.
- You can put the main () function wherever you want in your C program.
- However, the execution of your program always starts with the main () function.

1. 7 The first C Program

The main Function :

- The main () function body starts in line 4 and ends in line 6.
- Because this is a very simple program, the main () function is the only function defined in the program.
- Within the main () function body, a C library function, printf (), in line 5 is called in order to print out a greeting message.

1. 7 The first C Program

The Newline Character (\n):

- The newline character, \n, in line 5, usually suffixed at the end of a message.
- The newline character tells the computer to generate a carriage return and line feed sequence so that anything printed out after the message will start on the next new line on the screen.

1. 7 The first C Program

The return statement:

- All functions in C can return values.
- For instance, when you make a function to add two numbers, you can make such a function that returns to you the value of the addition.
- The main () function itself returns a value.
- By default, main () returns an integer.
- Therefore, in line 6 of the program, there is a statement, return 0; that indicates that 0 is returned from the main () function and the program is terminated normally.