CS1160
Lab 1: Introduction to C Programming

I. Overview

The objective of this lab is to introduce you to C programming and the basics of compiling a C application. Throughout this lab, you will learn how to compile and run your first C program, the main function the printing function.

II. What is C Programming?

C is a general-purpose, simple and flexible language that can be used to develop software like operating systems, databases, compilers.

'C' is a structured programming language in which program is divided into various modules. Each module can be written separately and together it forms a single 'C' program. This structure makes it easy for testing and maintaining processes.

C is a compiled language. A compiler is a special tool that compiles the program and converts it into the object file, which is machine-readable.

III. Getting Started

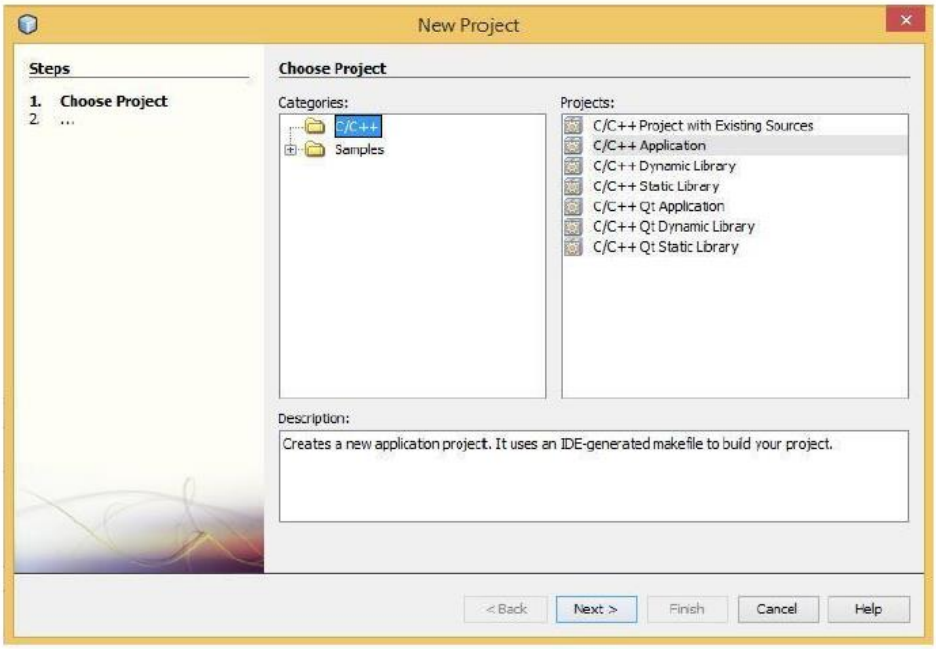
To Start writing your first C programming you need an IDE or **I**ntegrated **D**evelopment **E**nvironment.

The IDE is where you write your code, similar to the text editor, but it contains a compiler that build your code. In this lab you will use NetBeans.

Sara Abu Sa'aleek

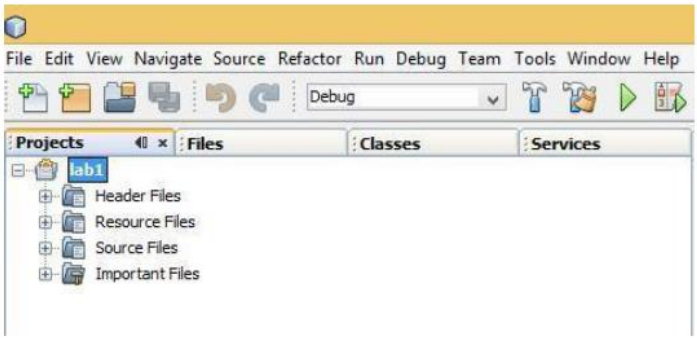
Creating a C/C++ Application Project:

1. Open the New Project wizard by choosing File > New Project.
2. In the wizard, select the C/C++ category.
3. The wizard gives you a choice of several types of new projects. Select C/C++ Application and click next.



4. Create a new C/C++ Application project from the wizard using the defaults. You can choose the name of the project and the location of the project.
5. Click Finish to exit the wizard.

A project is created with logical folders as follows:

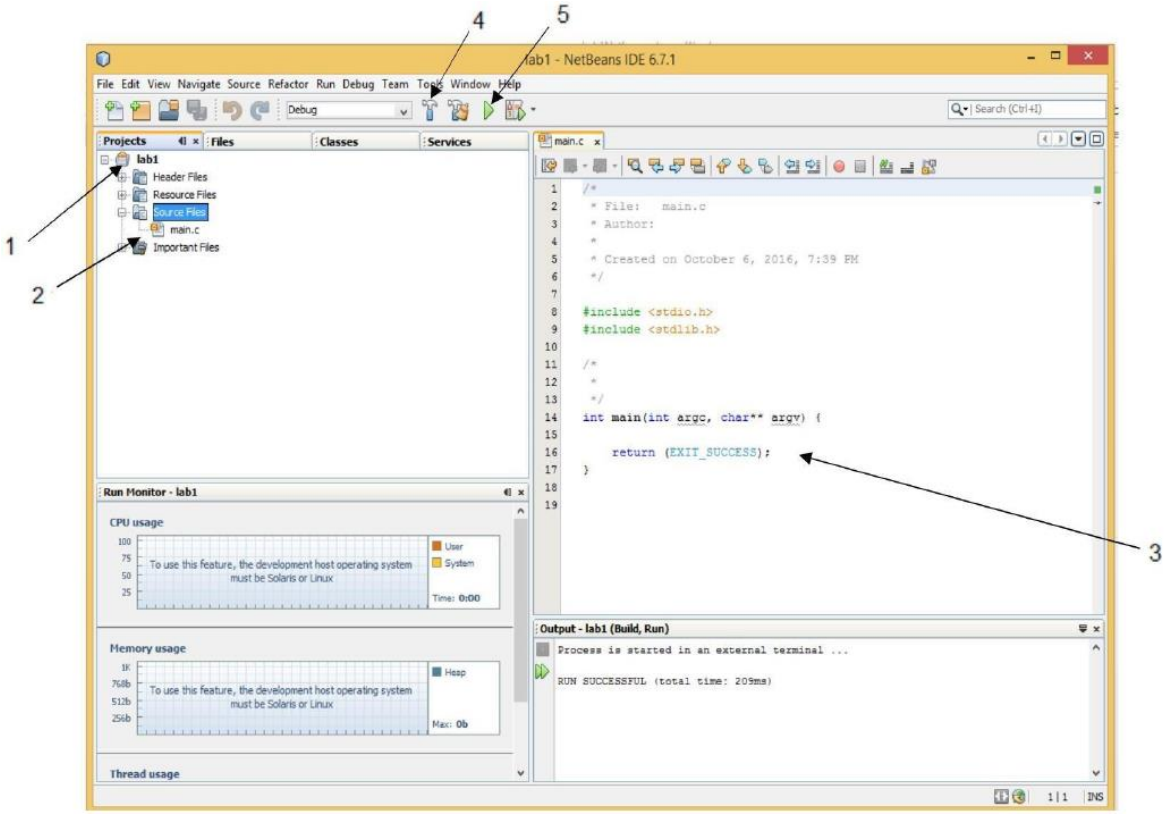


Navigating the NetBeans window:

Notice that the screen is divided into various areas. There are three (3) rows of information and three sub screens/windows.

The first row tells us among other things, the name of the Program. In this, case NetBeans IDE 6.7.1 (Yours might be an updated version say 7.3.1 or so).

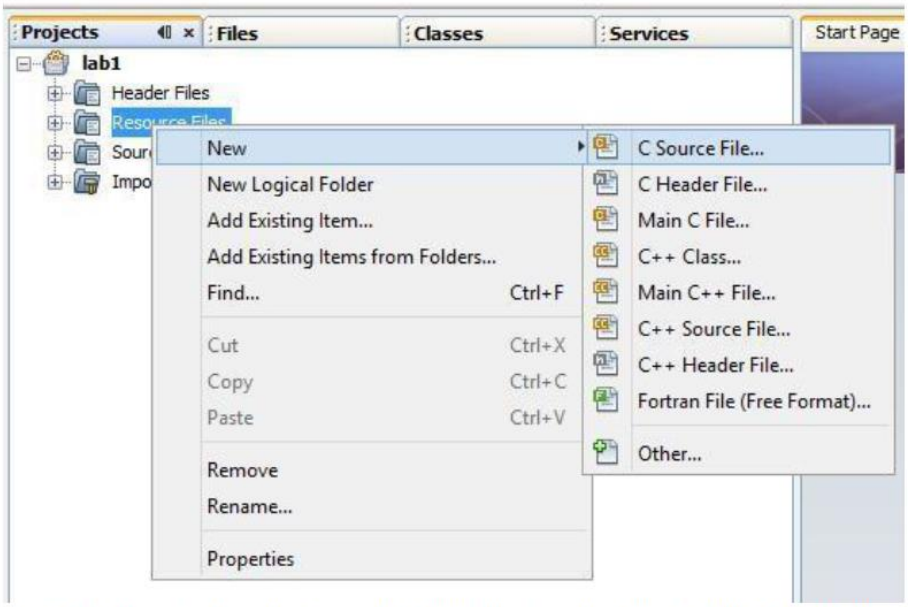
The second row is a menu showing the different tasks that you can perform. The first option is File. Click on it and you will see as in Figure 5 the different options for handling your C program file(s).



1. **The name of the project**
2. **The main file**
3. **Space to write your code**
4. **Compile tool**
5. **Run tool**

Adding New Files to Your Project

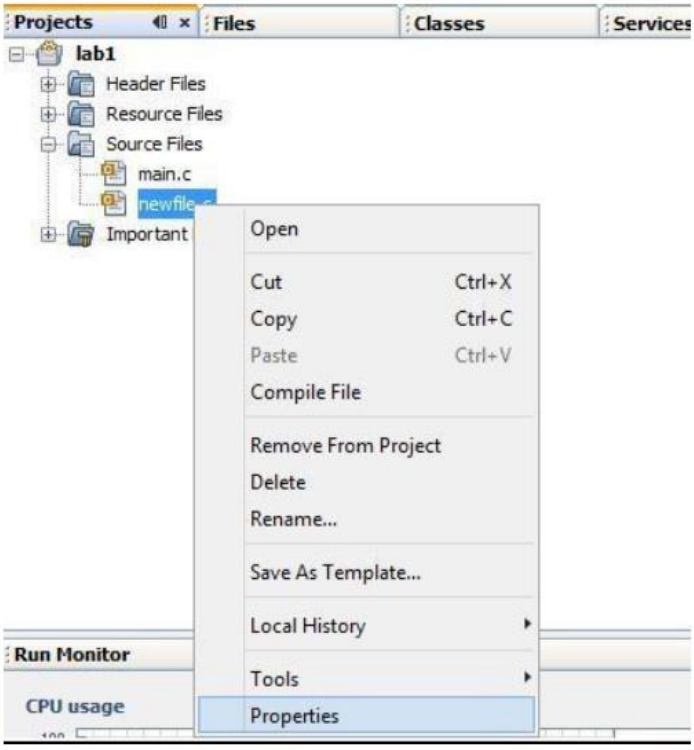
1. Right-click the Source Files folder and choose New > C Source File.



2. On the Name and Location page of the New File dialog box, type newfile in the File Name field.
3. Click Finish.

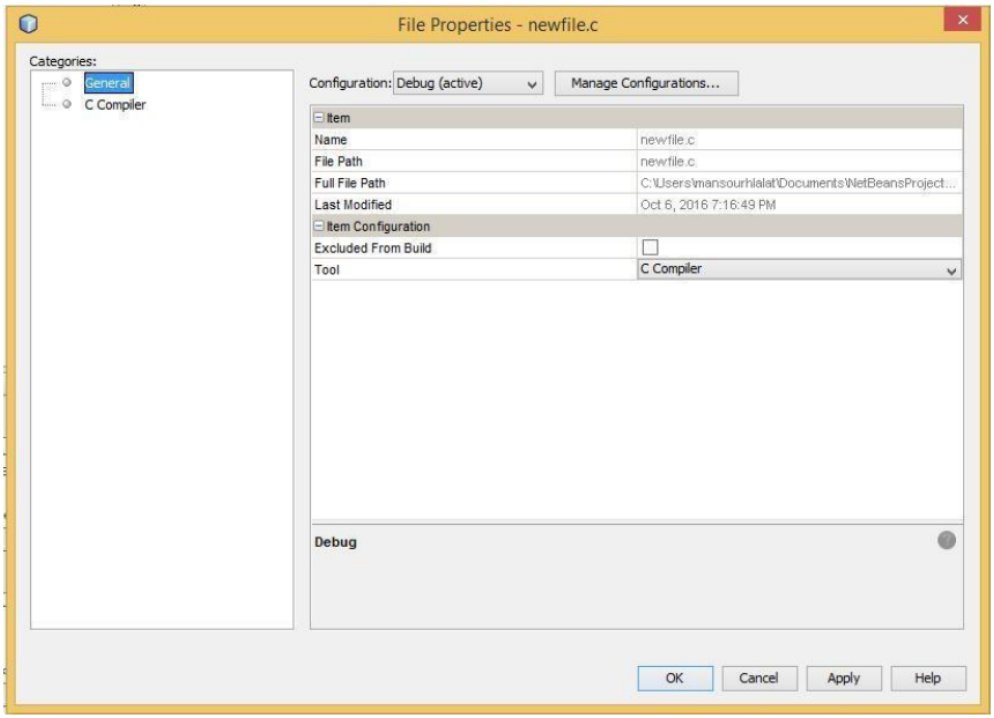
Setting Source File Properties

1. Right-click the newfile.c source file in the Projects tab and choose Properties.



-

2. Click the General category and see that you can specify a different compiler or other tool to build this file. You can also use a checkbox to exclude the file from the build of the currently selected project configuration.



IV. Your First Program

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

2. int main(void) {

3. printf("Hello World\n");

4. return 0;

5. }
```

Run the previous code and check the result.

- **#include <stdio.h>**
  This statement tells the compiler to use the (stdio) library; the name stands for standard input-output. This library gives us access to input/output functions, for instant the printf() function.

- **int main(void) {**
  The main() function is the entry point of any C program. A function is a routine that takes one or more arguments, and returns a single value. In the case of main(void), the function gets no arguments, and returns an integer. This is identified using the void keyword for the argument, and the int keyword for the return value.
  The function has a body, which is wrapped in curly braces. Inside the body, we have all the code that the function needs to perform its operations.

- **printf("Hello World\n");**
  The printf() function has no return value defined, because it is a pre-defined function in inside the stdio library, and it takes a string, wrapped in double quotes.

  \n is called an escape character that is used to go to a new line after the statement is printed.

- **return 0;**
  The function main() has no return value.

Here is some escape sequences used in C programming:

Escape Sequence	Meaning
\n	New Line
\t	Tab
\\	Backslash
\'	Single Quote
\"	Double Quote
\b	Backspace
\?	Question Mark

V. Task:

Write a C program that displays the days of your weekly class schedule along with the courses taught on each day. Use escape sequences such as \n, \t, and \" to format your output clearly.

- Sample of code:

```c
#include <stdio.h>

int main() {
    printf("Weekly Class Schedule:\n");
    printf("----------------------------\n");

    printf("Sunday\t\t: \"Programming I\"\tand \"Calculus I\"\n");
    printf("Monday\t\t: \"English Communication\"\tand \"Physics\"\n");
    printf("Tuesday\t\t: \"Programming I Lab\"\tand \"Discrete Mathematics\"\n");
    printf("Wednesday\t: \"Calculus I\"\t\tand \"Physics Lab\"\n");
    printf("Thursday\t: \"Computer Skills\"\tand \"Programming I\"\n");

    printf("\nNote:\\n is used for a new line and \\t for a tab space.\n");
    printf("Example of using quotes: \"C programming is fun!\"\n");

    return 0;
}
```

- Output:

```
Weekly Class Schedule:
----------------------------
Sunday          : "Programming I"       and "Calculus I"
Monday          : "English Communication"       and "Physics"
Tuesday         : "Programming I Lab"   and "Discrete Mathematics"
Wednesday       : "Calculus I"          and "Physics Lab"
Thursday        : "Computer Skills"     and "Programming I"

Note:\n is used for a new line and \t for a tab space.
Example of using quotes: "C programming is fun!"
```