ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

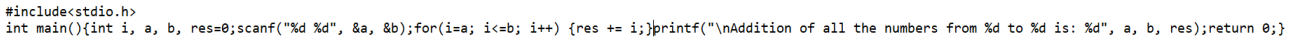
CSE 4108: Structured Programming I Lab
Lab 1

Objectives

- Getting familiar with program writing and execution environment.
- Necessity of writing a properly indented code.

Topics:

1. **Writing a Code:**
   You can write a C program in any basic text editor (even on built-in Windows Notepad). The following image is a screenshot of a C program written in Notepad. The source code of this program is attached in the google classroom.



   It is not difficult for a machine to execute this type of code, but for a human being, it is very difficult. Some advanced text editors (For example: sublime text, notepad++, VS-Code etc..) can highlight the keywords of the code and assist with auto indentation. Indentation is very helpful for a human being to understand the code.
   For this lab, proper indentation is **MANDATORY** while writing the code. Otherwise, your submission will be cancelled and sometimes, an additional penalty may be applied.

2. **Indentation:**
   There might be various conventions for the indentation of the code. The two most commonly used indentation styles are discussed below. Choose one of them and follow that convention throughout the whole semester (probably throughout your whole life). Choose wisely!!

   Indentation style 1:
   In this convention, you start the opening curly braces at the same line as the starting of the block. The closing curly braces are aligned with the starting of the block. All the lines inside the block are shifted by a TAB (some may prefer SPACE, but not recommended). A sample indented code following this style is provided below.

```c
#include<stdio.h>
int main(){
    int i, a, b, res=0;
    scanf("%d %d", &a, &b);
    for(i=a; i<=b; i++){
            res += i;
    }
    printf("\nAddition of all the numbers from %d to %d is: %d", a, b, res);
    return 0;
}
```

Indentation style 2:
In this convention, you will write both of the opening and closing curly braces aligned with the starting of the block. All the lines inside the block are shifted by a TAB (some may prefer SPACE, but not recommended). A sample indented code following this style is provided below.

```c
#include<stdio.h>
int main()
{
    int i, a, b, res=0;
    scanf("%d %d", &a, &b);
    for(i=a; i<=b; i++)
    {
        res += i;
    }
    printf("\nAddition of all the numbers from %d to %d is: %d", a, b, res);
    return 0;
}
```

Your task is to download the provided source code and format it with proper indentation following both styles described above and submit the formatted code in Google Classroom.

3. **Executing a program:**
To execute a C program, you need to have a compiler installed on your computer. Depending on the operating system, the installation process may vary. There are some software called Integrated Development Environment (IDE), which combines both text editor and compiler in one bundle (For example: Codeblocks, Visual Studio, Netbeans, etc..). Installing such software can make it easier to set up the environment. We have Codeblocks installed on all of the computers in our lab. If you need a different IDE or text editor, you may request the lab attendant or bring your own computer.

   For an IDE, usually, there is a button or a keyboard shortcut to compile and run a program easily. However, you can do it manually from the command prompt too. (The lab teacher will demonstrate how to compile and execute a C program in the command prompt). For the lab tasks, it is not mandatory to use the command prompt.

   Your task is to compile and execute all three source files (one is provided in the classroom, other two are your indented codes). Take a screenshot of your execution window and submit it in the Google Classroom. A sample screenshot is provided below:

4. **Adding comments:**
You can add comments to your code, which are lines of normal text within the code, to provide an explanation or add notes inside your program. The compiler ignores those lines and does not consider it as a part of your program. You can add a single-line comment using // and multi-line comments using /* (at the beginning), */ (at the end).

   Add your detailed information (Name, Student ID, Hometown, School and College name) inside any of the 3 source files created earlier and upload it to Google Classroom. Use both single-line and multi-line comments in your program.