Department of

Computer Science and Engineering

Title: Familiarization with the different programming languages, compiler, interpreter, assembler, IDE and Coding Environments

---

Computational Thinking and Problem Solving

CSE 100

Green University of Bangladesh

1. Objectives:

- Familiarize with different programming language.
- Differentiate between compiler, interpreter and assembler.
- Define IDE (Integrated development environment) and familiarize with coding environments.

2. Programming languages:

Language is a mode of communication that is used to **share ideas, opinions with each other**. For example, if we want to teach someone, we need a language that is understandable by both communicators.

A programming language is a **computer language** that is used by **programmers (developers) to communicate with computers**. It is a set of instructions written in any specific language (C, C++, Java, Python) to perform a specific task. A programming language is mainly used to develop desktop applications, websites, and mobile applications.

3. Types of programming language:

3.1. Low- level Programming Language:

Low-level language is **machine-dependent (0s and 1s)** programming language. The processor runs low- level programs directly without the need of a compiler or interpreter, so the programs written in low-level language can be run very fast.

Low-level language is further divided into two parts –

3.1.1. Machine Language-

Machine language is a type of low-level programming language. It is also called as **machine code or object code**. It does not require a translator to convert the programs because computers directly understand the machine language programs. It is a set of instructions that can be executed directly by the central processing unit (CPU) of an electronic computer. For instance, the following code is the binary executable code (i.e., a sequence of 0 and 1) defining a function (i.e., a kind of tool that takes some inputs and produces some output) for calculating the n^{th} Fibonacci number:

```
10001011010101000010010000001000100000111111101000000000011101110
00001101011100000000000000000000000000000000000000011000011100000
11111110100000001001110111000001101011100000000001000000000000000
00000000001100001101010011101110110000000100000000000000000000000
00001011100100000001000000000000000000000000010001101000001000
00110011000001111111010000000110111011000000111100010111101100110001
00111000001010010101110101111110001010110111100011
```

The advantage of machine language is that it helps the programmer to execute the programs faster than the high-level programming language.

3.1.2. Assembly language-

Assembly language (ASM) is also a type of low-level programming language that is designed for specific processors. It represents the set of instructions in a symbolic and human-understandable form. Even if it introduces humanly

recognizable symbols, one line of assembly code typically represents one machine instruction in machine language. It uses an assembler to convert the assembly language to machine language.

The advantage of assembly language is that it requires less memory and less execution time to execute a program.

3.2. High level programming Language:

High-level programming language (HLL) is designed for **developing user-friendly software programs and websites**. This programming language requires a compiler or interpreter to translate the program into machine language (execute the program).

The main advantage of a high-level language is that it is **easy to read, write, and maintain**.

High-level programming language includes **Python, Java, JavaScript, PHP, C#, C++, Objective C, Cobol, Perl, Pascal, LISP, FORTRAN, and Swift programming language**.

A high-level language is further divided into following parts –

3.2.1. Procedural Oriented programming language:

Procedural Oriented Programming (POP) language is derived from structured programming and based upon the procedure call concept. It divides a program into small procedures called routines or functions.

Procedural Oriented programming language is used by a software programmer to create a program that can be accomplished by using a programming editor like IDE, Adobe Dreamweaver, or Microsoft Visual Studio.

The advantage of POP language is that it helps programmers to easily track the program flow and code can be reused in different parts of the program.

Example: C, FORTRAN, Basic, Pascal, etc.

3.2.2. Object-oriented programming Language:

Object-Oriented Programming (OOP) language is **based upon the objects**. In this **programming language, programs are divided into small parts called objects**. It is used to implement real-world entities like inheritance, polymorphism, abstraction, etc. in the program to makes the program reusable, efficient, and easy-to-use.

The main advantage of object-oriented programming is that OOP is faster and easier to execute, maintain, modify, as well as debug.

Example: C++, Java, Python, C#, etc.

3.3. Middle-level programming language

Middle-level programming language lies between the low-level programming language and high-level programming language. It is also known as the intermediate programming language and pseudo-language.

A middle-level programming language's advantages are that it supports the features of high-level programming, it is a user-friendly language, and closely related to machine language and human language.

Example: C, C++, language.

4. Compiler:

In computing, a compiler is a computer program that translates computer code written in one programming language (the source language) into another language (the target language). The name "compiler" is primarily used for programs that translate source code from a high-level programming language to a low-level programming language (e.g. assembly language, object code, or machine code) to create an executable program.

5. Interpreter:

An interpreter is a computer program that is used to directly execute program instructions written using one of the many high-level programming languages.

The interpreter transforms the high-level program into an intermediate language that it then executes, or it could parse the high-level source code and then performs the commands directly, which is done line by line or statement by statement.

6. Assembler:

An assembler is a type of computer program that interprets software programs written in assembly language into machine language, code and instructions that can be executed by a computer.
An assembler enables software and application developers to access, operate and manage a computer's hardware architecture and components.

7. Integrated Development Environment:

IDE stands for Integrated Development Environment. It is a programming environment that contains a lot of things in a single package i.e., code editor, compiler, debugger. It is actually a software application that provides comprehensive facilities to computer programmers for software development. It combines all the basic tools that developers need to write or test software. This type of environment allows an application developer to write code while compiling, debugging and executing it at the same place. It can be a standalone application or a part of one or more compatible applications.

Example: CodeBlocks, NetBeans.

8. CodeBlocks:

Code Blocks is a great tool for beginners who want to get started coding or for those looking for a way to improve their skills. It has tons of features and a user-friendly interface that makes learning coding easy. CodeBlocks is a cross-platform IDE that supports compiling and running multiple programming languages.

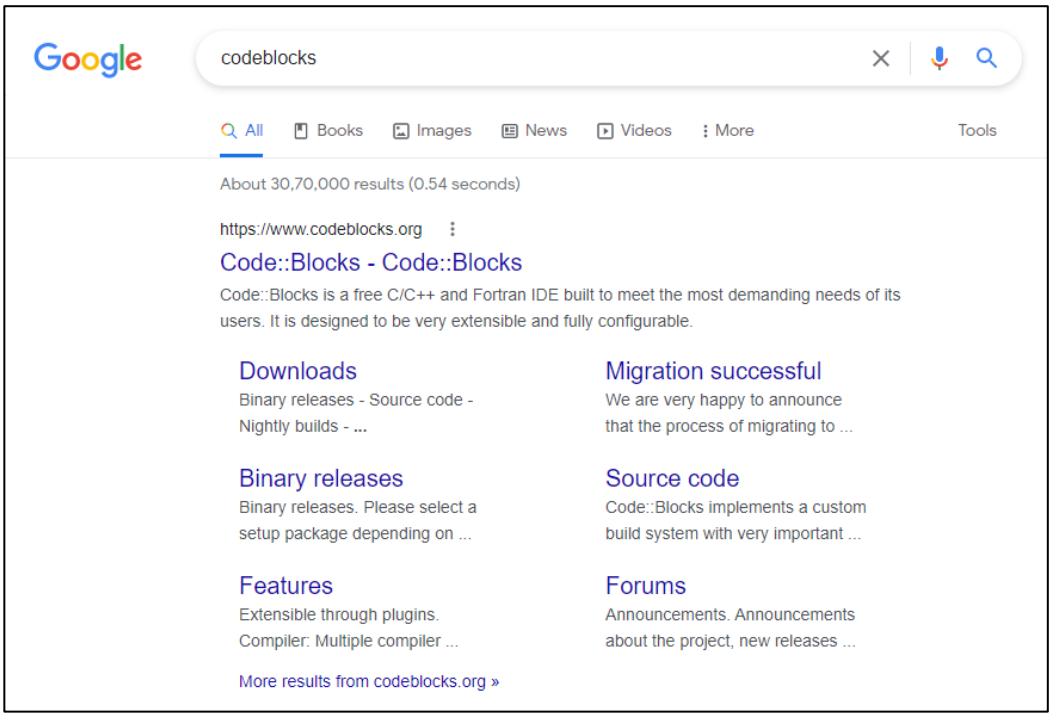
8.1. Installation Code Blocks for C++ on Windows:

Follow the below steps to install Code Blocks for C++ on windows:

Step 1: Open Your Web Browser

Step 2: Go to the Search Panel and Search for "Code Blocks"

Step 3: Click on the First Result shown on the Search Engine or click on this link.



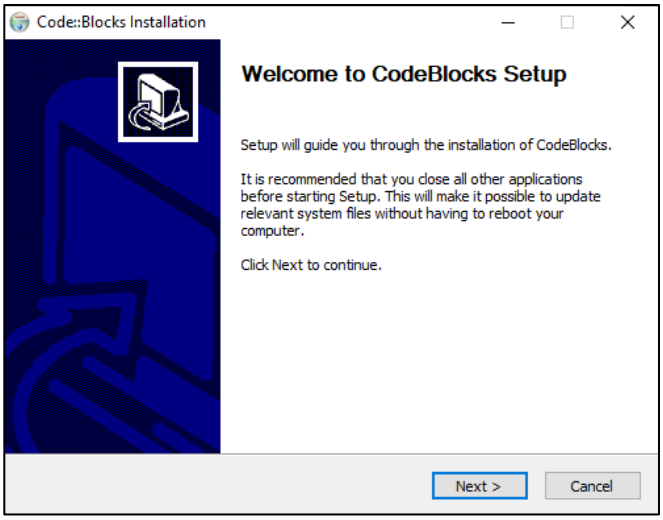
Step 4: Click on the "Downloads" Section.

Step 5: Click On "Download the binary release".

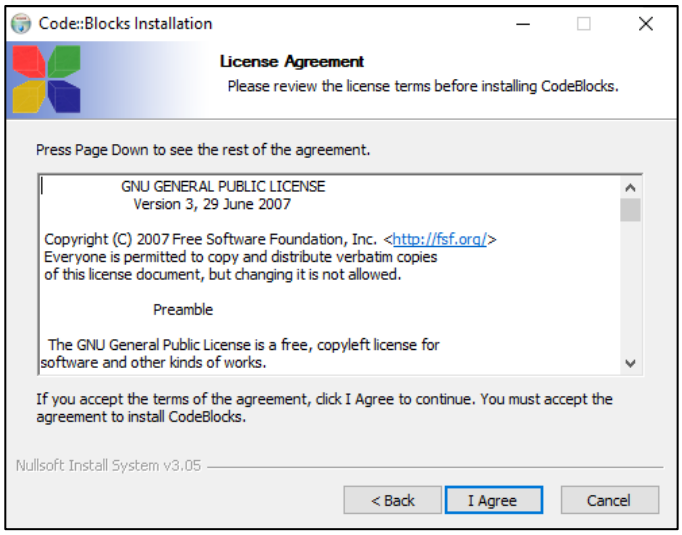
Step 6: Now here you'll find several download options. As per this Date, download the latest version. You must download the file with MinGW written on it as MinGW is a Compiler that is needed to run the Program. If you download the normal setup file then you have to download the compiler separately.

Step 7: When the download is completed, Open Your Code Blocks Setup File.

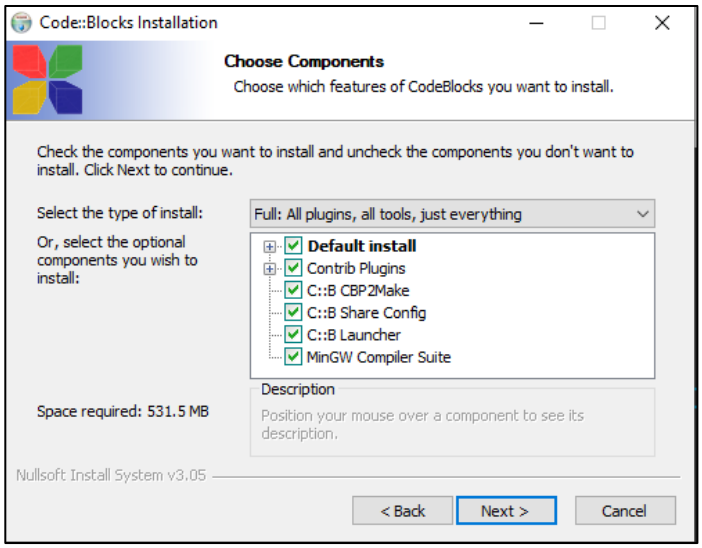
Step 8: Click on Next.



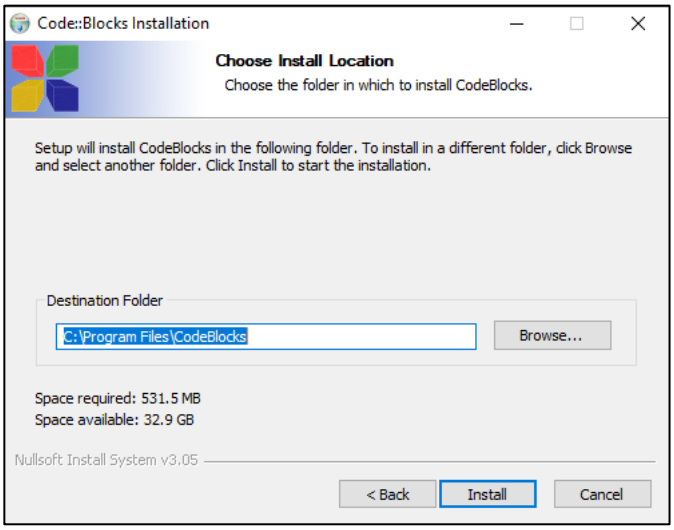
Step 9: Click on I agree.

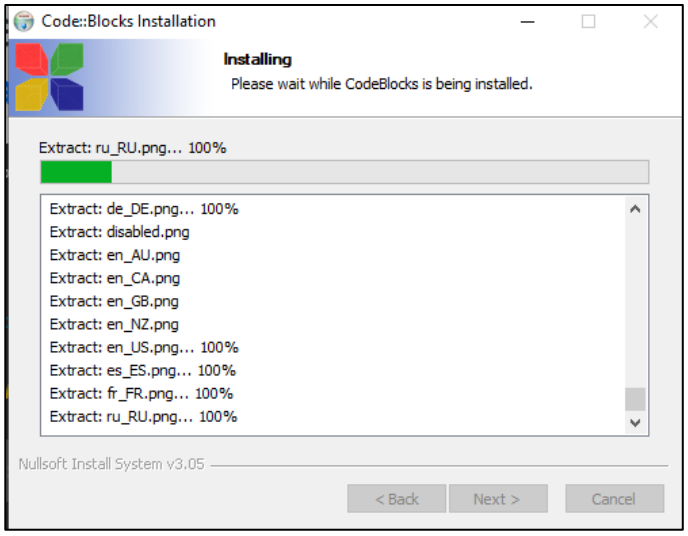


Step 10: Click on Next.

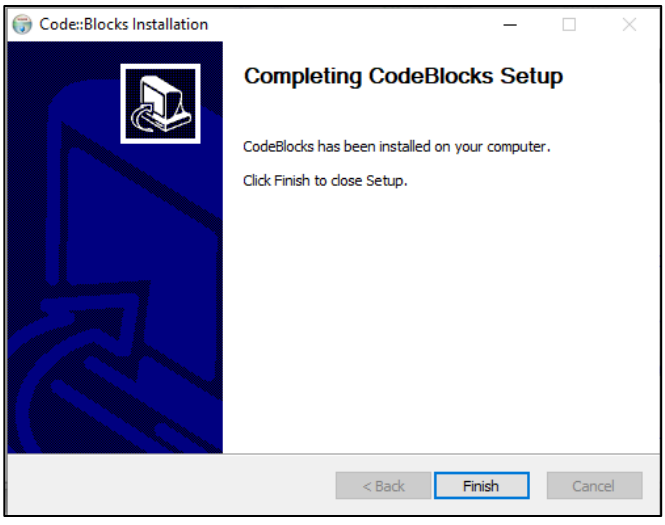


Step 11: Select your Destination and Click on Install.





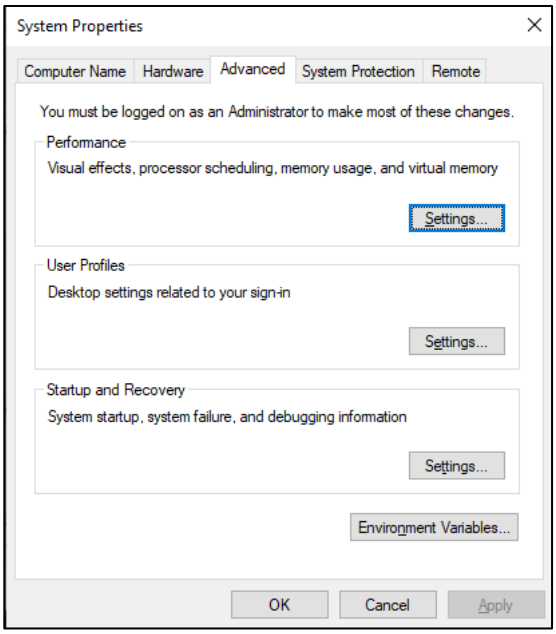
Step 12: Once Installation gets completed, click on Next and then Click on Finish.



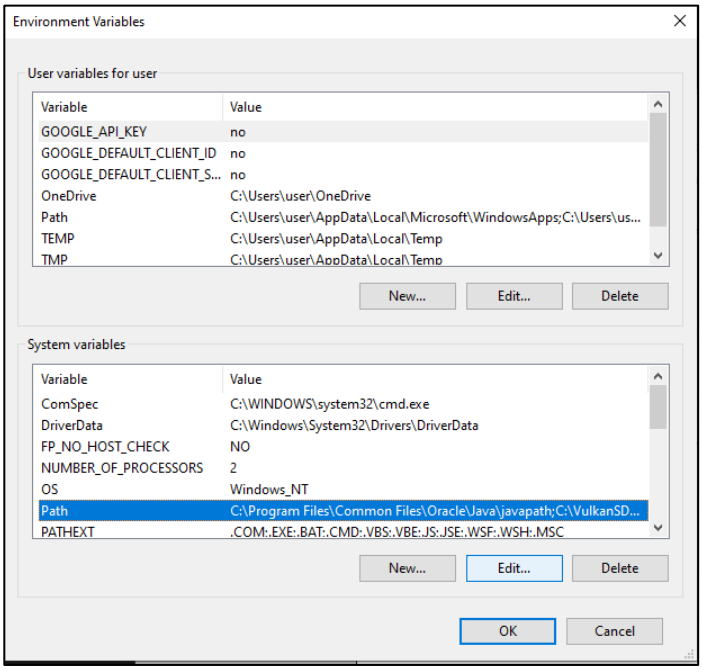
Now You Code Blocks have been installed.

To Set the Environment Path of GCC compiler

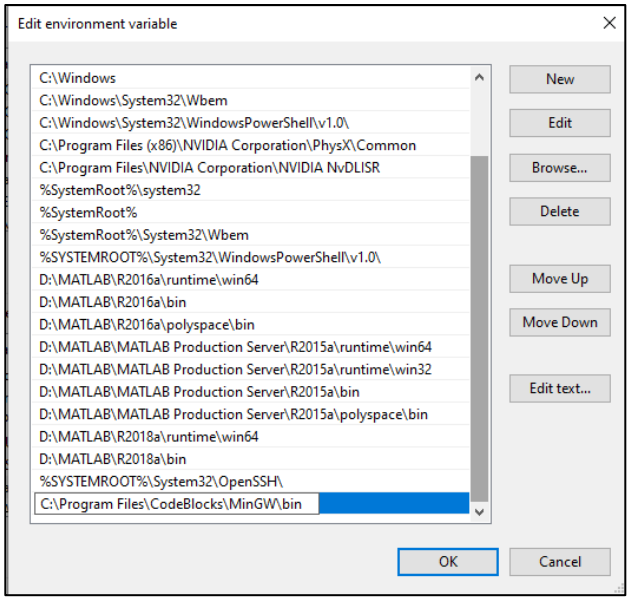
Step 1: Go to your Code Blocks MinGW installation folder location (For me it is C:\Program Files\CodeBlocks\MinGW\bin) and copy the address
Step 2: Go to Search Panel and type "**Edit System environment variable**s"
Step 3: Click on Environment Variables



Step 4: Under System variables, Click on Path and Select Edit



Step 5: Click on New and Paste the Address into it and click on OK



Now Code Blocks will automatically detect GCC Compiler.

Lab Task (Please implement yourself and show the output to the instructor)

1) Implement a c program to print “Hello world” and explore the 3 different files generated by the IDE (Source file, object file, executable file).

Lab exercise (submit as a report)

1) Installation of CodeBlocks IDE in your personal computer.
2) Make a report demonstrating the different menus of codeblocks.

Policy

Copying from the internet, classmates, seniors, or from any other source is strongly prohibited. 100% marks will be deducted if any such copying is detected.