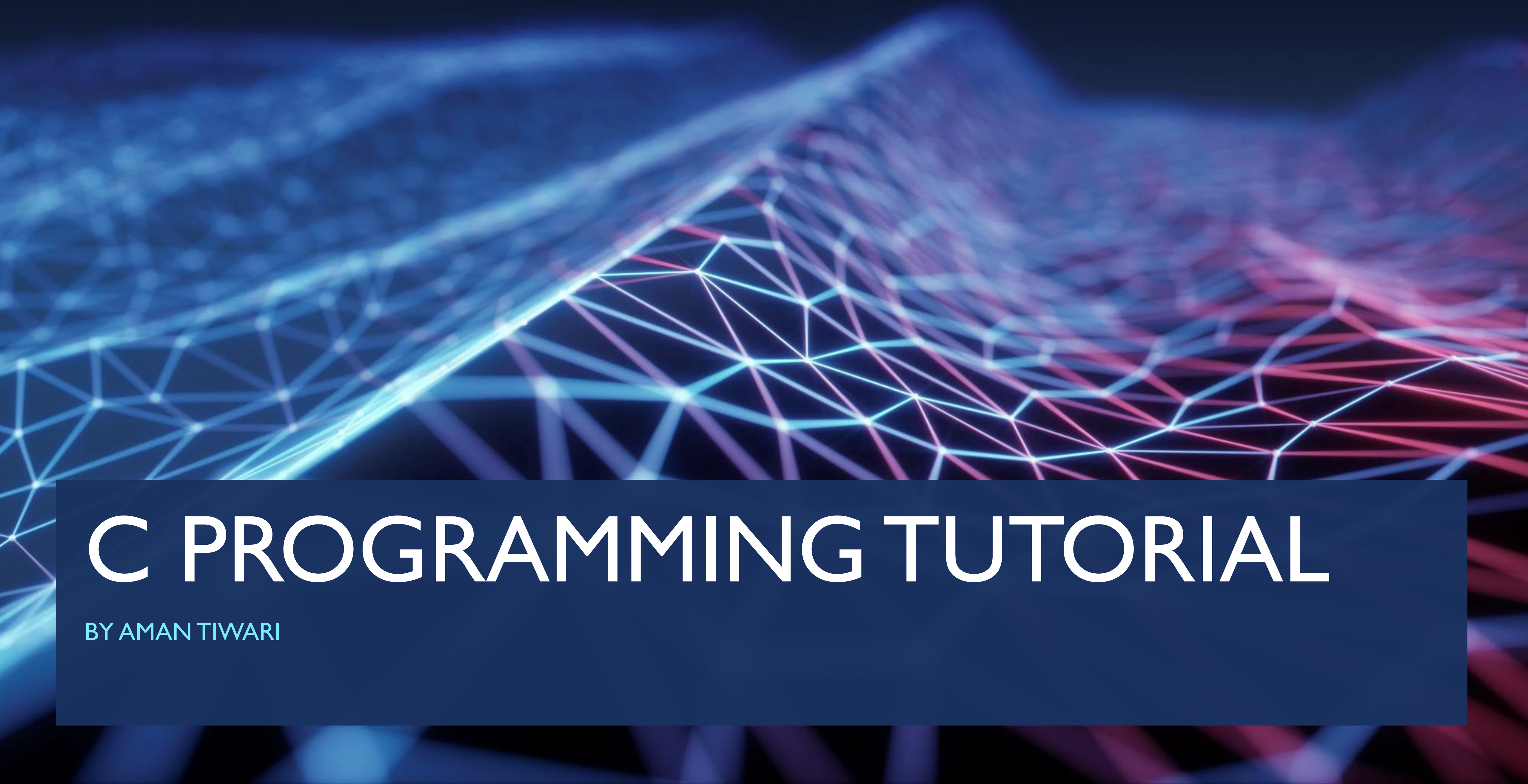


C PROGRAMMING TUTORIAL

BY AMAN TIWARI

What is a Programming Language

A programming language is a set of instructions that can be used to interact with and control a computer. These languages are used to design websites, create apps, develop operating systems, control spacecraft, and analyze data. Programming languages are necessary because computers can't understand Engish. Programming languages bridge this gap by helping programmers translate their commands into something that the computer can understand and execute.

Computers are made of many tiny switches that can be either on or off. When a switch is on, it is represented by a 1. When it is off, it is represented by a 0. These 1s and 0s are called **bits**. Bits are the fundamental language of nearly all computers and every program must be translated into bits before it can be executed by the computer. When 8 bits are grouped together, this is called a **byte**. A byte can represent a letter, for example, 01100001 represents 'a'. A byte can also represent a control character.
For example, 00000011 signals the end of a piece of text. When representing a number using bits, it can be converted from its normal base 10 representation to binary. This is called **binary** representation.

Types of Programming Languages

There are three types of programming languages: machine language, assembly language, and high-level language. **Machine language(low level)** is easier for the computer to understand but harder for the programmer to understand. This is because machine language is simply the language of machines—bits. Sometimes, programmers will develop programs directly with machine code, but because this is difficult to understand and tedious to type, it is more common to program using assembly or a high-level language.

Assembly language(mid level) is slightly easier to understand. The bits of machine language are replaced by numbers and English commands. Before assembly code is run by the computer, it is assembled by an **assembler**. This converts the code back into the 1s and 0s of machine language that the computer can understand.

High-Level languages use many more English commands and are significantly more readable than assembly or machine language. Many high-level languages have built-in commands that help the programmer write loops, create variables of different data types, and manipulate strings. It is worth noting that all of these are possible in assembly or machine language, but high-level languages make them much easier for the programmer to read, write, and debug. Some newer high-level languages are **scripting languages**. This means that they are not **compiled**, or translated into machine language, until just before the code is executed at runtime. Python, Javascript, PHP, Ruby, and Bash are all scripting languages.

PROGRAMMING LANGUAGES

Low level	Mid Level	High Level
They are machine dependent which directly access the machine resources using bits , CPU registors, pointers,etc	It contains 50% features of low level and 50% Features of high level. Eg. C & C++	They are machine independent and they Follows another methodology like OOPs (object oriented programming). Eg. Java ,python C# etc

History of C language is interesting to know. Here we are going to discuss a brief history of the c language.

C programming language was developed in 1972 by Dennis Ritchie at bell laboratories of AT&T (American Telephone & Telegraph), located in the U.S.A.

Dennis Ritchie is known as the **founder of the c language**.

WHAT IS C AND WHY C ?

The C Language is developed for creating system applications that directly interact with the hardware devices such as drivers, kernels, etc.

It provides feasible interaction with hardware devices without compromising on the performance.

C programming is considered as the base for other programming languages, that is why it is also known as mother language.

It can be defined by the following ways:

1. Mother language
2. System programming language
3. Procedure-oriented programming language
4. Structured programming language
5. Mid-level programming language

1) C as a mother language

C language is considered as the mother language of all the modern programming languages because **most of the compilers, JVMs, Kernels, etc. are written in C language**, and most of the programming languages follow C syntax, for example, C++, Java, C#, etc.

2) C as a system programming language

A system programming language is used to create system software. C language is a system programming language because it **can be used to do low-level programming (for example driver and kernel)**. It is generally used to create hardware devices, OS, drivers, kernels, etc. For example, Linux kernel is written in C.

3) C as a procedural language

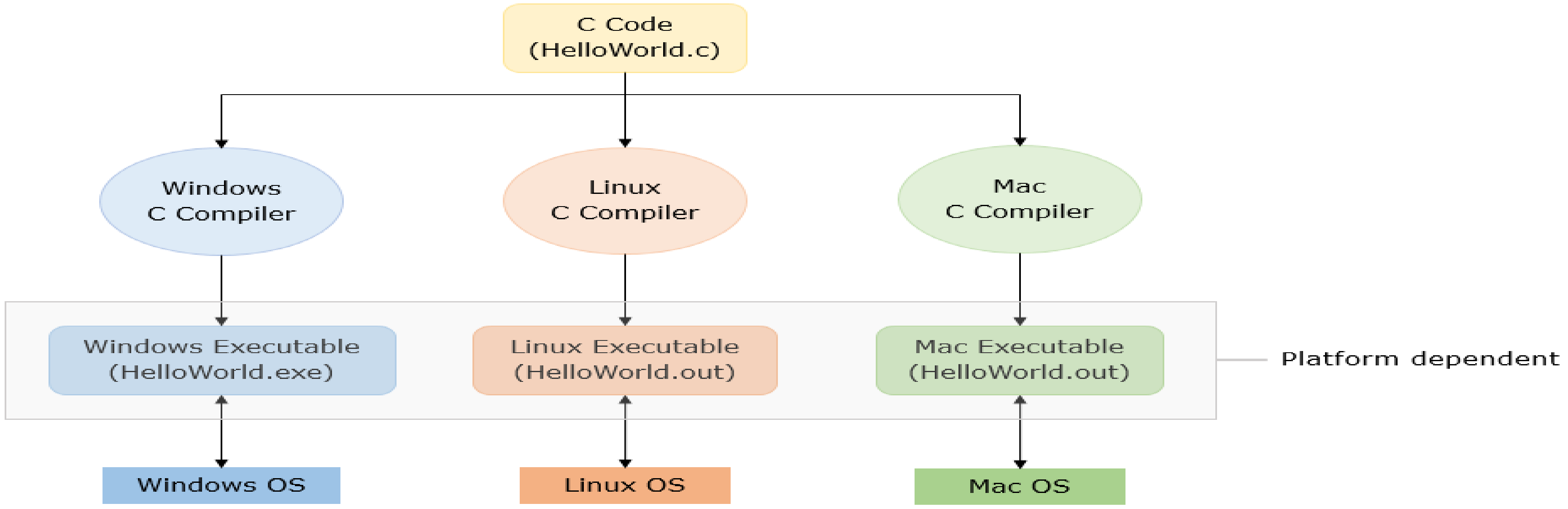
A procedure is known as a function, method, routine, subroutine, etc. A procedural language **specifies a series of steps for the program to solve the problem**.
A procedural language breaks the program into functions, data structures, etc.
C is a procedural language. In C, variables and function prototypes must be declared before being used.

4) C as a structured programming language

A structured programming language is a subset of the procedural language. **Structure means to break a program into parts or blocks** so that it may be easy to understand.
In the C language, we break the program into parts using functions. It makes the program easier to understand and modify.

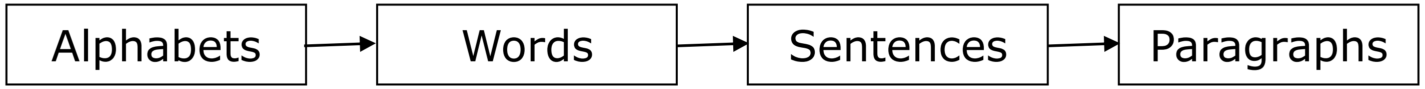
WHY C IS NOT PLATFORM INDEPENDENT

- Sometimes, it means the same as "hardware dependent" or "machine dependent" and refers to applications that run in only one hardware series with the operating system not being relevant. In contrast, **"platform independent" means that the application can run in different operating environments**.

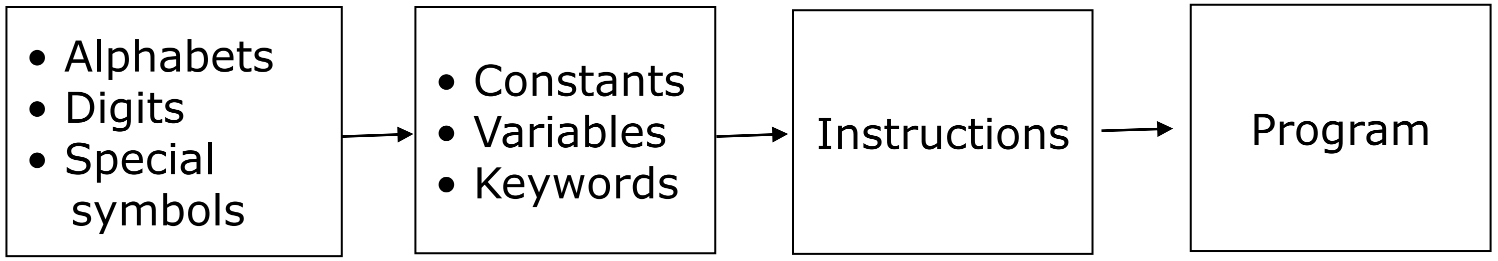


LET'S START WITH C

- Steps in learning English language:



- Steps in learning C:



Basic Structure of C Program In modern Compilers

```
#include <stdio.h>      // standard input output header file
int main()    // Starting point of a program
{   // Starting body of main
    printf("Hello in C Programming");   // Printing output on console
    // Semicolon(;) means end of line or statement(instruction)
    return 0;     //Ending point of a program
}  // Ending of body
```

Basic Structure of C Program In Turbo C/C++

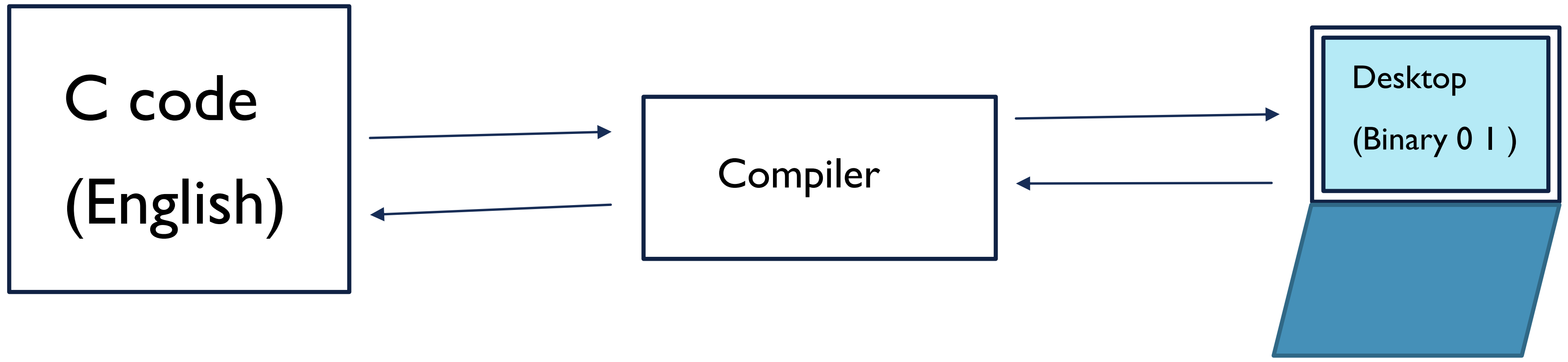
```
#include <stdio.h>
#include<conio.h>
int main()
{
    clrscr();
    printf("Hello C Programming\n");
    getch();
    return 0;
}
```

C KEYWORDS

- Keywords represents words whose meaning have been already explained to the C Compiler.
- Keywords cannot be used as variable names.
- Keywords are also called ‘Reserved words’.
- 32 keywords available in C are:

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
cont	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

ENVIRONMENT SETUP



Text Editor :-

Eg. Notepad,Notepad++, Sublime,atom,brackets, Vs code etc.

Compiler :-

GCC (GNU compiler collection) ,bcc,MinGW, onleinegdb compiler etc.

Device (Terminal) :-

command prompt, powershell,Terminal etc.

IDE : Integrated Development Environment   (ide = text editor + compiler + terminal + intellisense etc)

Eg. Turbo C/C++ ,code blocks, dev c/c++ , eclipse ,visual studio (> 20 GB) etc.

My Combination : VS code + GCC + powershell /cmd

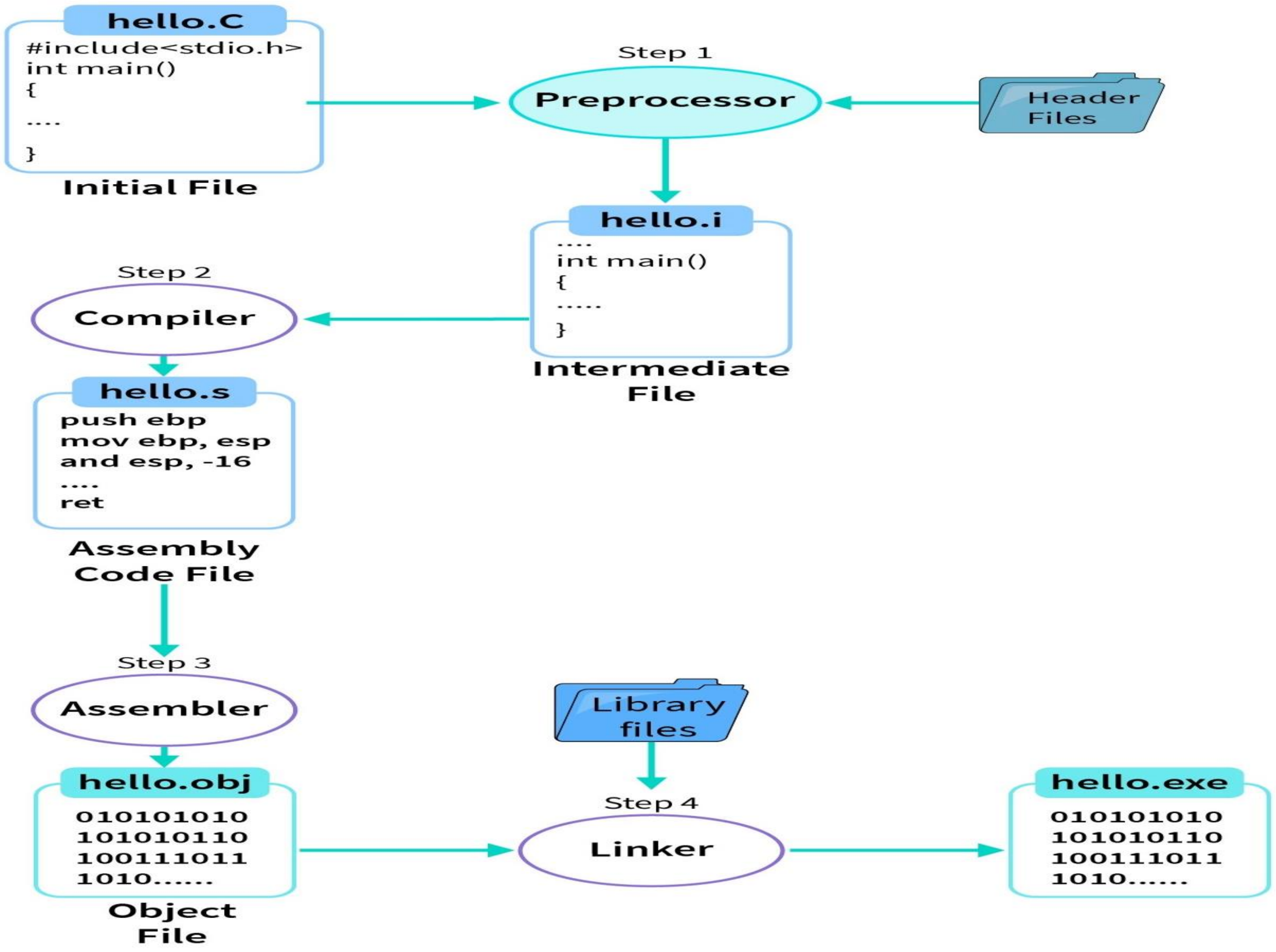
For Begginers : dev c/c++ ,code blocks

https://jmeubank.github.io/tdm-gcc/download/

https://code.visualstudio.com/download

https://sourceforge.net/projects/dev-cpp/files/Binaries/Dev-C%2B%2B%204.9.9.2/devcpp-4.9.9.2_setup.exe/download

https://sourceforge.net/projects/codeblocks/files/latest/download


hello.C
#include<stdio.h>
int main()
{
....
}
Initial File
Step 1
Preprocessor
Header Files
hello.i
....
int main()
{
.....
}
Intermediate File
Step 2
Compiler
hello.s
push ebp
mov ebp, esp
and esp, -16
....
ret
Assembly Code File
Step 3
Assembler
hello.obj
010101010
101010110
100111011
1010......
Object File
Library files
Step 4
Linker
hello.exe
010101010
101010110
100111011
1010......

THANK YOU

AMANTIWARI8861@GMAIL.COM