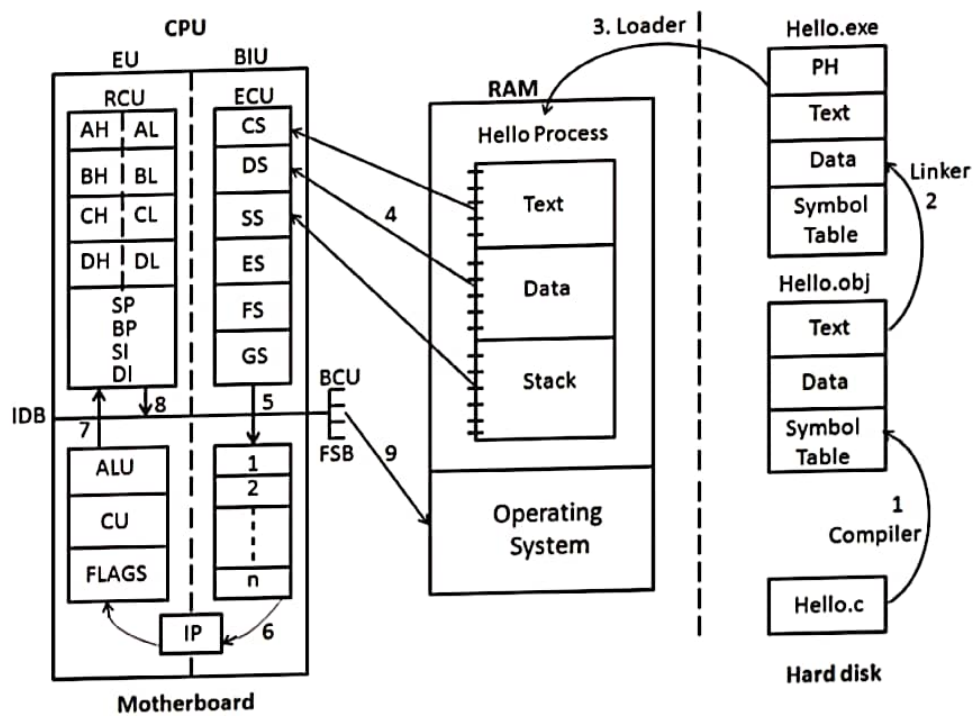


Computer Architecture

A Toolchain is the set of Programming Tools that are used to create software.
The Tools are used in a chain, so that the output of each tool becomes input for the next tool.



Components of Toolchain

Editor :

Editor is program in which we can write our code and edit that code. Gedit, kwrite are the examples of editors.

File Name: Demo.c

```
#include<stdio.h>
#define MAX 10
int Add(int No1, int No2)
{
    int Ans;
    Ans = No1 + No2 + MAX ;
    printf("Addition of Two Numbers is %d",Ans); returnAns;
}
```

Marvellous Infosystems : Pre-Placement Activity



Preprocessor:

The Preprocessor provides the ability for the inclusion of header files, macro expansion, conditional compilation, and line control. Preprocessor takes lines beginning with '#' as directives.

Because it knows nothing about the underlying language. After Preprocessing it will give the expanded code as a output having extension .i. Expanded output given by preprocessor is in human understandable format.

File Name: Demo.i

```
int printf( const char * format, ...);
int scanf( const char * format, ...);
int Add(int No1, int No2)
{
    int Ans;
    Ans = No1 + No2 + 10;
    printf("Addition of Two Numbers is %d",Ans);
    returnAns;
}
```

Compiler:

Compiler is a program which is used for translating source code from a high level programming language (like c,c++) to lower level programming language (like assembly language, machine language). In this case we consider that output is in assembly language having extension .asm or .s.

Code which is generated by the compiler is machine dependent.

File Name: Demo.asm

```
Add:  PUSH ECX PUSH EDX
      ADD ECX,EDX
      ADD ECX,10
      MOV EAX,ECX
      RETN
```

Assembler:

Assembler is a program which translates assembly language program to an object file, which contains code in machine language.

Object file contains machine code but still it is non-executable having extension .obj.

File name: Demo.obj

```
10111011101110101110110 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101010
10110110111000101001010 10101110100010100101010 10101110100010100101010
```

Linker:

A Linker is a program that takes one or more object files and combines them into a single executable program having extension .exe.

Linker adds primary header over executable file which contains address of entry point function, magic number, time-date stamp, type of executable (Executable can be self-executable or dependable executable).

File name: Demo.exe

Demo.obj

```
10111011101110101110110 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101010
10110110111000101001010 10101110100010100101010 10101110100010100101010
+
```

Other.obj

```
10111011101110101110110 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101010
10110110111000101001010 10101110100010100101010 10101110100010100101010
=
```

Demo.exe

```
10111011101110101110110 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101010
10110110111000101001010 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101010
```

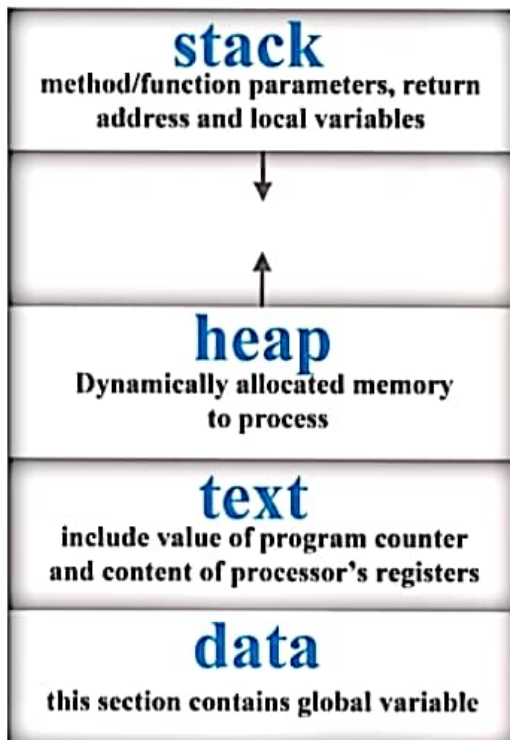
Loader:

A Loader is a program which is part of operating system that is responsible for loading programs. It places program in memory (RAM) and prepares them for execution. Loading a program involves reading the contents of executable file like text, data into the memory.

Demo.exe running in RAM

```
10111011101110101110110 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101010
10111011101110101110110 10101110100010100101010 10101110100010100101011
```

Marvellous Infosystems : Pre-Placement Activity



When our program gets loaded into memory it becomes process and divided into three parts as Text, Data, and Stack.

Text section is a section which contains compiled instructions of our program in machine understandable format.

Data section contains all the global variable and static variables of our program. Data section is internally divided into two parts as bss(Block starting with Symbol) and non bss(Block starting with value).

bss section contains all non-initialized global variables and non bss section contains all initialized global variables.

Stack is a data structure which is used to maintain all the information about the functions as local variables, input arguments, Address of next instruction to be executed, old EBP etc.

Set of commands which will elaborate the Toolchain components

Step 1: Create file names as Demo.c which contains simple C program.

Step 2: Compile that file using command
gcc -S -o Demo.S Demo.c

Step 3: Pass that Assembly file to assembler which gives output as a object file named as Demo.o
as -o Demo.o Demo.S

Step 4: Pass that object file to Linker which gives output as a executable
ld -o Demo -lc -dynamic-linker /lib/ld-linux.so.2 Demo.o -e main

Step 5: Run that executable using
./Demo

Programming Languages

It is an artificial language designed to communicate with a computer. Programming languages can be used to create programs that control the behaviour of a computer. The description of a programming language is usually split into the two components of syntax and semantics. Syntax means rules of writing the program and semantics means rules of meaning.

Types of Programming Language

Procedural Programming Languages

Procedural programming specifies a list of operations that the program must complete to reach the desired state. This approach is also known as imperative programming. Integral to the idea of procedural programming is the concept of a procedure call. Procedures, also known as functions, subroutines, or methods, are small sections of code that perform a particular task. Ex. FORTRAN and BASIC.

Structured Programming Languages

Structured programming is a special type of procedural programming. It provides additional tools to manage the problems that larger programs were creating. Structured programming requires that programmers break program structure into small pieces of code that are easily understood.

It also frowns upon the use of global variables and instead uses variables local to each subroutine. Ex. C, Ada, and Pascal.

Object-Oriented Programming Languages

Object-oriented programming (OOP) is a programming paradigm that represents concepts as "objects" that have data fields (attributes that describe the object) and associated procedures known as methods. Objects, which are usually instances of classes, are used to interact with one another to design applications and computer programs. Ex. Java, C++

Subscribe to the Marvellous Infosystems YouTube channel for getting the notifications.

Marvellous Infosystems YouTube Channel Link : <https://www.youtube.com/c/marvellousinfosystems>

For more information please call Admin : 7020713938