

Introduction to Scientific and Engineering Computation (BIL 104E)

Lecture 1

Introduction to C programming

Introduction

- **As engineers and scientists why do we need computers?**

A computer is a device that can perform computations and make logical decisions billions of times faster than human beings can.

- **Why do we need computer program ?**

A computer is a machine that performs operations that are specified with a set of instructions called a program.

- **What is C ?**

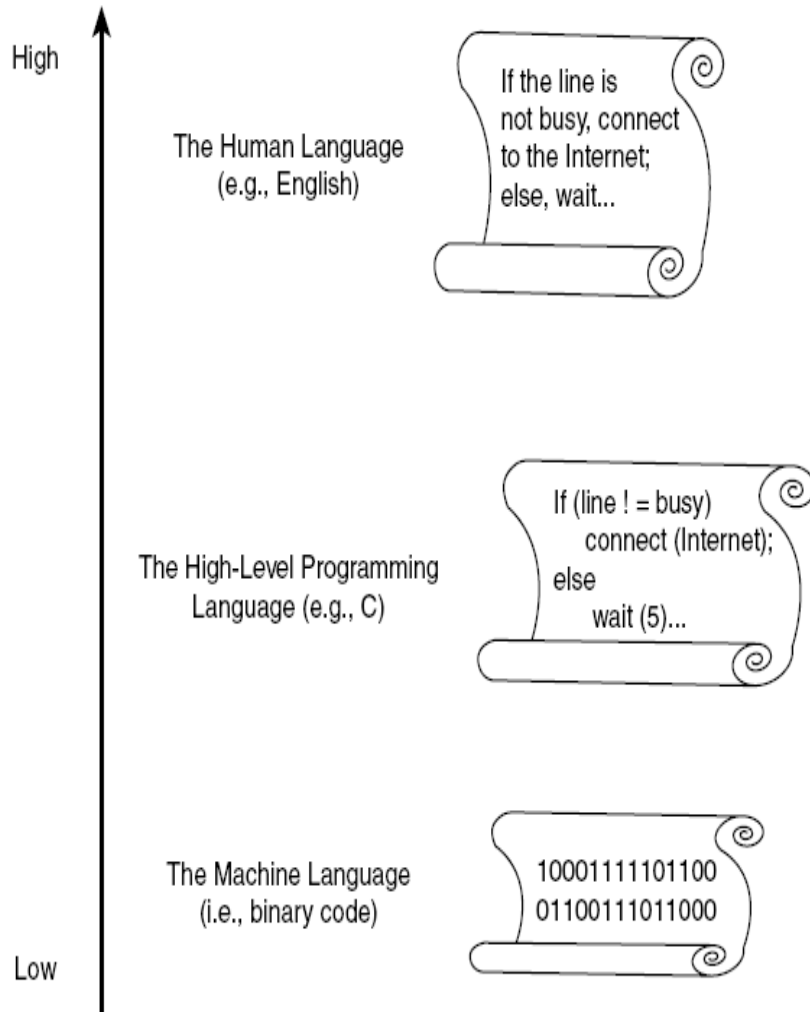
C is a high level programming language. The C language was first developed in 1972 by Dennis Ritchie at AT&T Bell Labs.

- **Why C ?**

C is one of the most popular general purpose programming language.

C can easily be used for system-level operations, that's why it is called medium level programming language.

Computer Languages



1-) Machine Language:

that is defined by its hardware design is written using two symbols which are usually represented using the digits 0 and 1 (binary).

2-) Assembly Language:

is also unique to a specific computer design. But instructions are written in English-like statements instead of binary.

3-) High – level Language:

have English-like commands and instructions. Include C, FORTRAN etc. Writing programs is easier.

High-level Languages

Advantages of high-level languages including C:

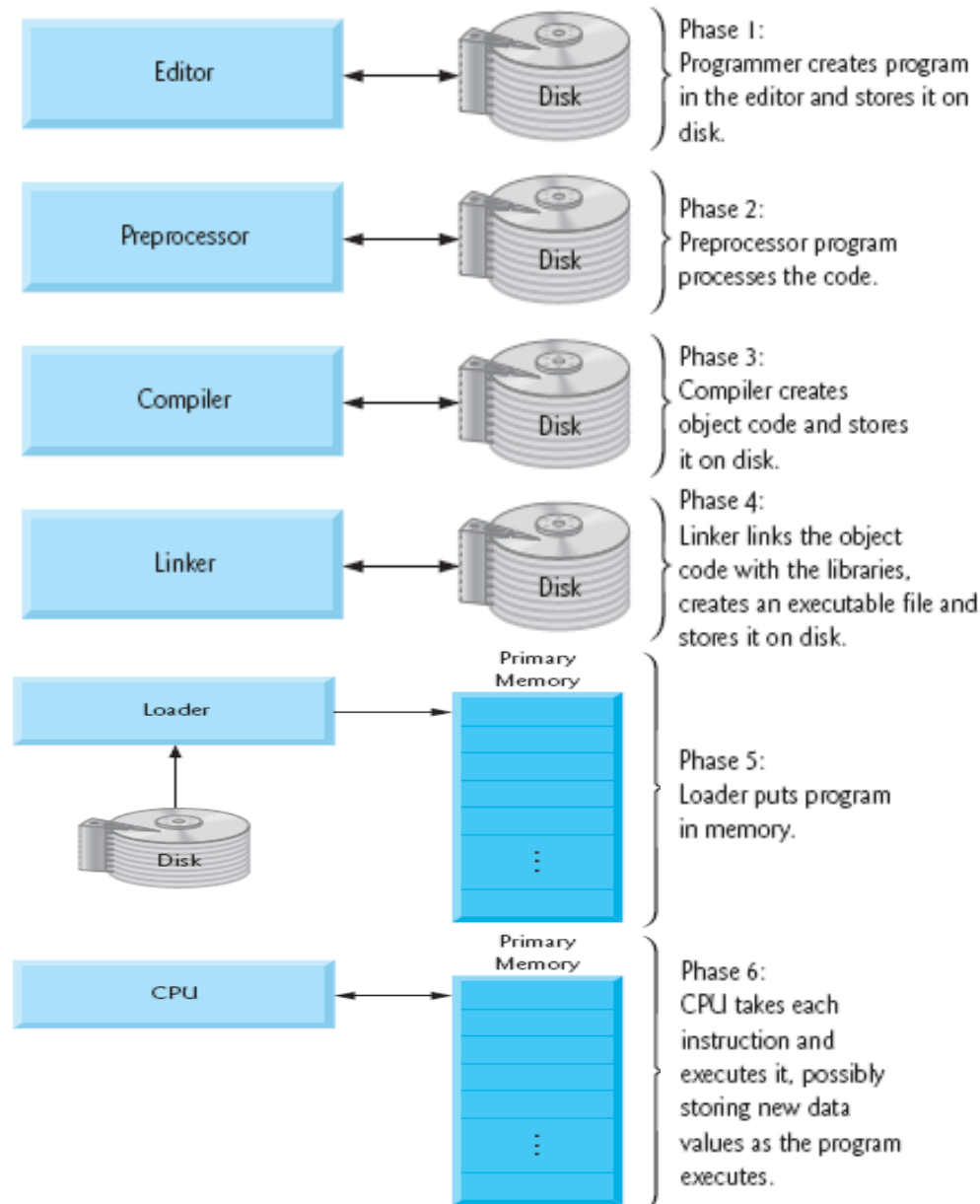
Readability : Programs are easy to read.

Maintainability: Programs are easy to maintain.

Portability: Programs are easy to port across different computer platforms.

Each high-level programming language needs a compiler (simply interpreter) to translate instructions into machine language

C Development Environment



Problem Solving Methodology

1. State the problem clearly
2. Describe the input and output information
3. Work a simple example by hand
4. Develop an algorithm and convert it to a computer program
5. Test the solution with a variety of data

Example

Problem statement:

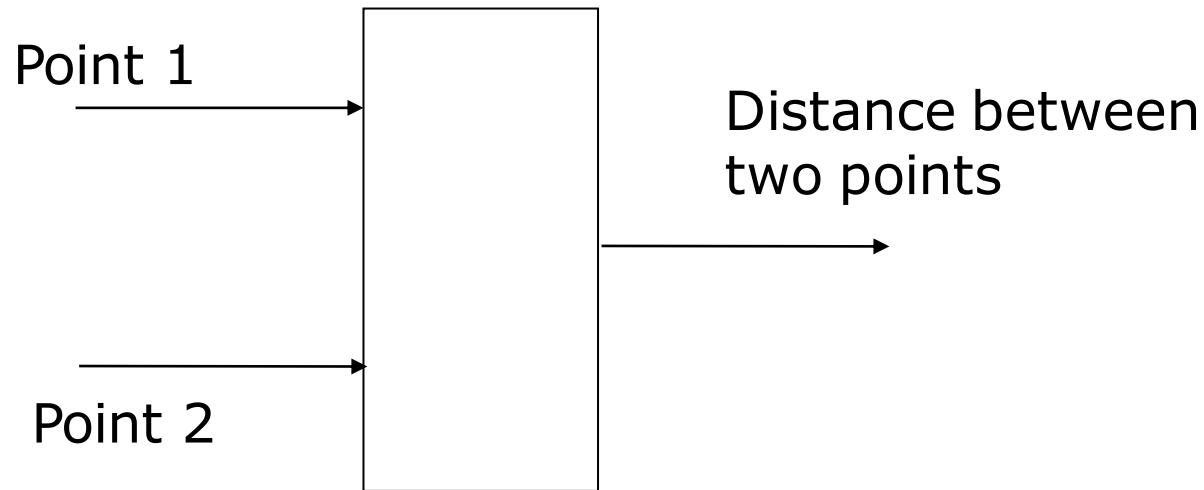
- Should be clear and avoid any misunderstanding.

“Compute the distance between two points in a plane.”

Example

Input/Output Description :

- Information that is given and the values to be computed.



Example

Hand Example :

$$p_1 = (1,5)$$

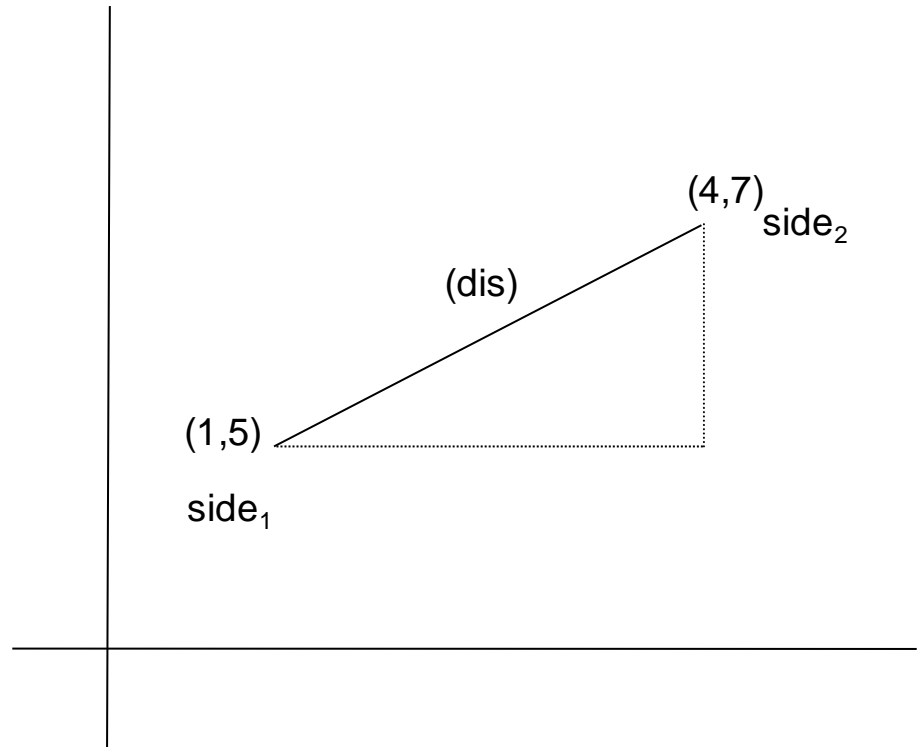
$$p_2 = (4,7)$$

$$dist = \sqrt{(side_1)^2 + (side_2)^2}$$

$$= \sqrt{(4-1)^2 + (7-5)^2}$$

$$= \sqrt{13}$$

$$= 3.61$$



Example

Algorithm Development :

- Algorithm is a step-by-step outline of the problem solution, in other words simple operations performed one after another.

1. Give values to the points.
2. Compute the lengths of the sides of the right triangle. (side1 and side2).
3. Compute the hypotenuse of the triangle which is also the distance between the two points. (distance)
4. Print the value of distance.

Example

```
/*--This program computes the distance between two points-----*/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
main()
{
    /*----- Declare and initialize variables----- */
    double x1=1, y1=5, x2=4, y2=7;
    double side1, side2, dist ;

    /*----- Compute sides of the right triangle-- */
    side1=x2-x1;
    side2=y2-y1;
    dist=sqrt(side1*side1+side2*side2);

    /*----- Print distance on the screen----- */
    printf("The distance between the two points is %5.2f \n",dist);

    /*----- Exit program ----- */
    return EXIT_SUCCESS;
}
```

Example

Testing :

- Data from the hand example can be used for testing.

The output of the program is:

The distance between the two points is 3.61