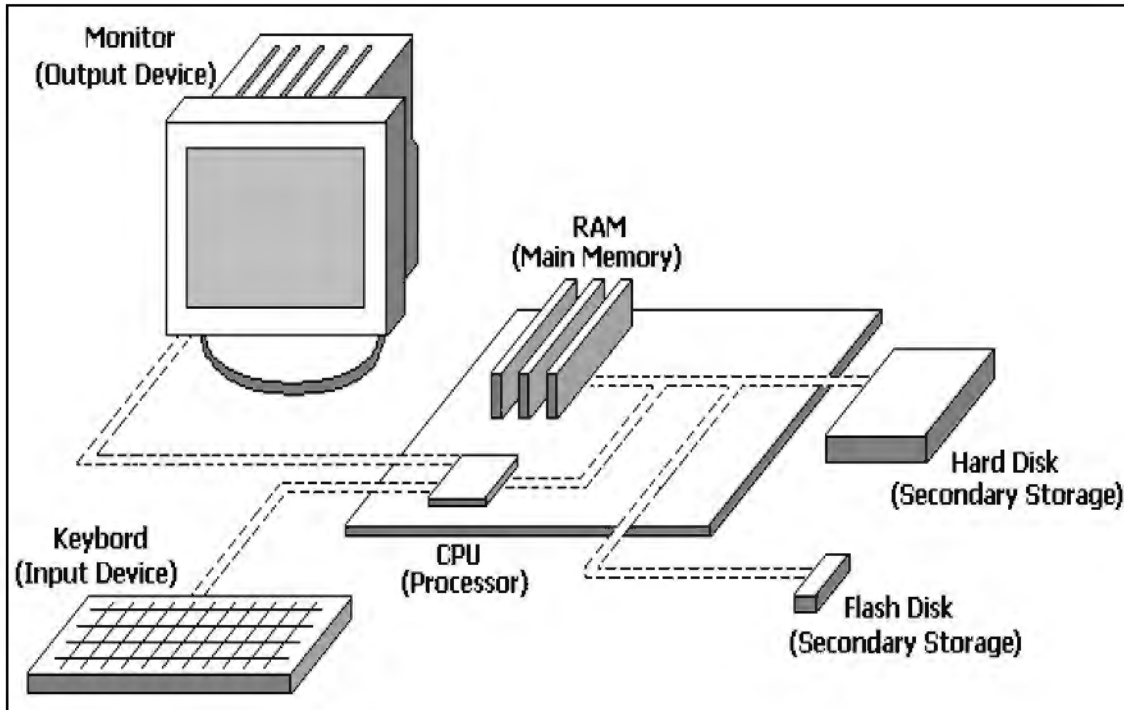


LECTURE 1

1. Introduction:



hardware components

Computer is a device capable of performing computations and making logical decisions at speeds millions and even billions of times faster than human beings.

Computers process data under the control of sets of instructions called **computer programs**.

Programming is the process of writing instructions for a computer in a certain order to solve a problem.

The computer programs that run on a computer are referred to as **software (SW)**. While the hard component of it is called **hardware (HW)**.

Developing new software requires written lists of instructions for a computer to execute. Programmers rarely write in the **language** directly understood by a computer.

2. Short History:

The following is a short history, just for given a general view of how languages are arrived:

- 1954: Fortran.
- 1957: Cobol.
- 1958: Algol (*Base for Simula*).
- 1958: Lisp.
- 1961: B1000.
- 1962: Sketchpad.
- 1964: Basic.
- 1967: Simula67.
- 1968: FLEX.
- 1970: Pascal (*From Algol*).
- **1971: C** (*From a language called B*).
- 1972: Smalltalk72 (*Based on Simula67 and Lisp*).
- 1976: Smalltalk76.
- 1979: ADA (*From Pascal*).
- **1980: C with classes** (*experimental version*).
- **1983: C++ (by Bjarne Stroustrup).**
- 1986: Objective-C (*from C and Smalltalk*).
- 1986: Eiffel (*from Simula*).
- 1991: Sather (*From Eiffel*).
- 1991: Java.
- 2000: C#.



Bjarne Stroustrup
at: AT&T Labs

3. C++ Programming Language:

For the last couple of decades, the C programming language has been widely accepted for all applications, and is perhaps the most powerful of structured programming languages. Now, C++ has the status of a structured programming language with object oriented programming (OOP).

C++ has become quite popular due to the following reasons:

1. It supports all features of both structured programming and OOP.
2. C++ focuses on function and class templates for handling data types.

4. C++ Program Development Process (PDP):

C++ programs typically go through six phases before they can be executed. These phases are:

1. **Edit:** The programmer types a C++ source program, and makes correction, if necessary. Then file is stored in disk with extension (.cpp).
2. **Pre-Processor:** Pre-processing is accomplished by the pre-processor before compilation, which includes some substitution of files and other directories to be include with the source file.
3. **Compilation:** Converting the source program into object-code.
4. **Linking:** A linker combines the original code with library functions to produce an executable code.
5. **Loading:** The loader loads the program from the disk into memory.
6. **CPU:** Executes the program, residing in memory.

These steps are introduced in the figure below:

