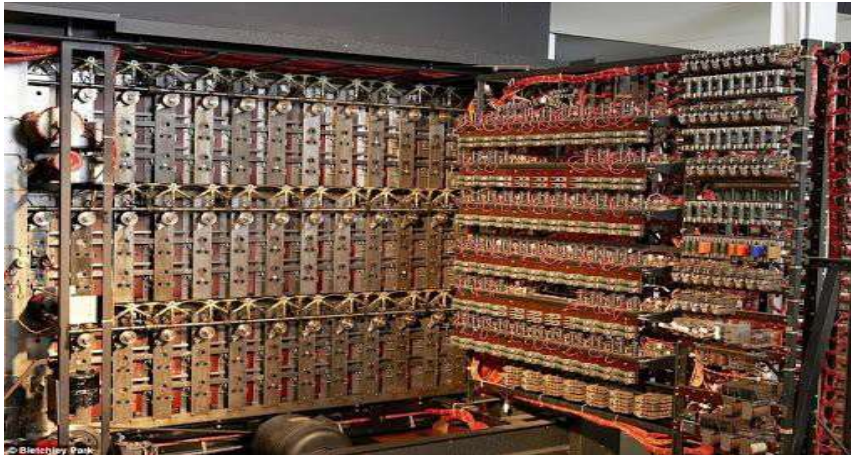




## Lecture 1: Introduction to Computer

### Historical background

“Alan Turing, who cracked the German military's secret codes by building the world's first computer.”



### What is a computer?

The computer is an electronic device that stores retrieve and processes data in **binary form** according to our needs. It takes raw data as input from the user, process it, and produces some output.

### Functionalities of a computer:

If we look at it in a very broad sense, any digital computer carries out the following five functions –

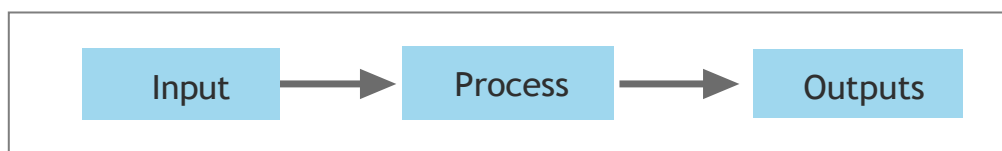
**Step 1** – Takes data as input.

**Step 2** – Stores the data/instructions in its memory and uses them as required.

**Step 3** – Processes the data and converts it into useful information.

**Step 4** – Generates the output.

**Step 5** – Controls all the above four steps.

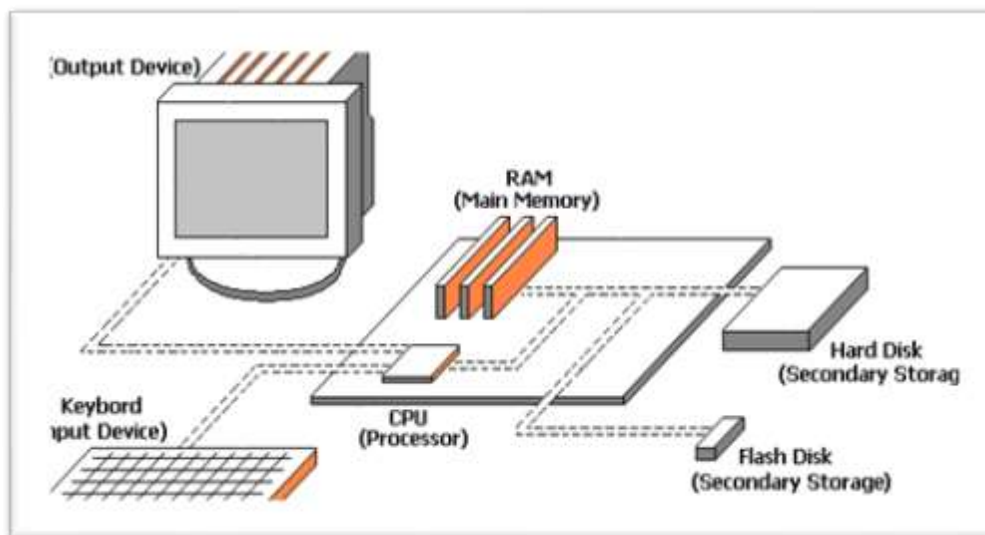


## Computer Generations:

Generation in computer terminology is a change in technology a computer is/was being used. Initially, the generation term was used to distinguish between varying hardware technologies. Nowadays, generation includes both hardware and software, which together make up an entire computer system.

There are **five computer generations** known till date. Each generation has been discussed in detail along with their time period and characteristics. Following are the main five generations of computers.

Generation	Description
First Generation	The period of the first generation: 1946-1959. Vacuum tube based.
Second Generation	The period of the second generation: 1959-1965. Transistor-based.
Third Generation	The period of the third generation: 1965-1971. Integrated Circuit based.
Fourth Generation	The period of the fourth generation: 1971-1980. VLSI (Very Large-Scale Integration) microprocessor based.
Fifth Generation	The period of the fifth generation: 1980-onwards. ULSI microprocessor based.



Hardware components

### A basic computer consists of four components:

1. Input device (or input unit e.g., optical, radar, ultrasound, sound sensor, microphones).
2. CPU (Central Processing Unit).
3. Memory (or Memory unit).
4. Output device (Output unit, text, speakers, images, .....).

## What is C++?

- C++ is a cross-platform language that can be used to create high-performance applications.
- C++ was developed by Bjarne Stroustrup, as an extension to the C language.
- C++ gives programmers a high level of control over system resources and memory.

The language was updated 4 major times in 2011, 2014, 2017, and 2020 to C++11, C++14, C++17, and C++20.

## Why Use C++

- C++ is one of the world's most popular programming languages.
- C++ can be found in today's operating systems, Graphical User Interfaces, and embedded systems.
- C++ is an object-oriented programming language which gives a clear structure to programs and allows code to be reused, lowering development costs.
- C++ is portable and can be used to develop applications that can be adapted to multiple platforms.
- C++ is fun and easy to learn!

## 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.

## 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 corrections, if necessary. Then the file is stored on disk with an 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 included 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.

**CPU:** Executes the program, residing in memory. These steps are introduced in the figure below:

