

PROGRAMMING FOR PROBLEM SOLVING --

PPS

SUBJECT CODE-
BTPS-101-18



PPS SYLLABUS

Unit 1

Introduction to Programming (4 lectures)

Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.) – **(1 lecture)**.

Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm: Flowchart/Pseudocode with examples. **(1 lecture)**

From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code- **(2 lectures)**

PPS SYLLABUS

Unit 2

Arithmetic expressions and precedence (**2 lectures**)

Conditional Branching and Loops (**6 lectures**)

Writing and evaluation of conditionals and consequent branching (**3 lectures**)

Iteration and loops (**3 lectures**)

Unit 3

Arrays (6 lectures)

Arrays (1-D, 2-D), Character arrays and Strings

PPS SYLLABUS

Unit 4

Basic Algorithms (6 lectures)

Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required)

PPS SYLLABUS

Unit 5

Function (5 lectures)

Functions (including using built in libraries), Parameter passing in functions, call by value, Passing arrays to functions: idea of call by reference

Unit 6

Recursion (4 -5 lectures)

Recursion, as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc. Quicksort or Merge sort.

PPS SYLLABUS

Unit 7

Structure (4 lectures)

Structures, Defining structures and Array of Structures

Unit 8

Pointers (2 lectures)

Idea of pointers, Defining pointers, Use of Pointers in self-referential structures,
notion of linked list (no implementation)

Unit 9

File handling (only if time is available, otherwise should be done as part of the lab)



UNIT-1

INTRODUCTION TO PROGRAMMING

CONTENTS

- 1. Introduction to components of a computer system**
- 2. Idea of Algorithm: steps to solve logical and numerical problems.**
- 3. Representation of Algorithm: Flowchart/Pseudocode**
- 4. From algorithms to programs**
- 5. Data types, variables and memory locations**
- 6. Syntax and Logical Errors in compilation**
- 7. Object and executable code**

WHAT IS PROGRAMMING

**In Which
language am I
speaking?**



WHAT IS PROGRAMMING

The language in which we communicate with the computer is called as programming language



WHAT IS PROGRAMMING

Programming Languages



WHAT IS COMPUTER

We will be
learning C
language in our
syllabus



WHAT IS COMPUTER

Computer is an electronic device for storing and processing data, typically in binary form, according to instructions given to it in a variable program.

Computer.. Latin word.. Compute
Calculation Machine



PROPERTIES OF COMPUTER

Capabilities of Computers

- Huge Data Storage
- Input and Output
- Processing

Characteristics of Computers

- High Processing Speed
- Accuracy
- Reliability
- Versatility
- Diligence

WHAT IS COMPUTER

- **Accepts input, processes data, stores data, and produces output**
- **Input** refers to whatever is sent to a Computer system
- **Data** refers to the symbols that represent facts, objects, and ideas
- **Processing** is the way that a computer manipulates data
- A computer processes data in a device called the central processing unit (CPU)

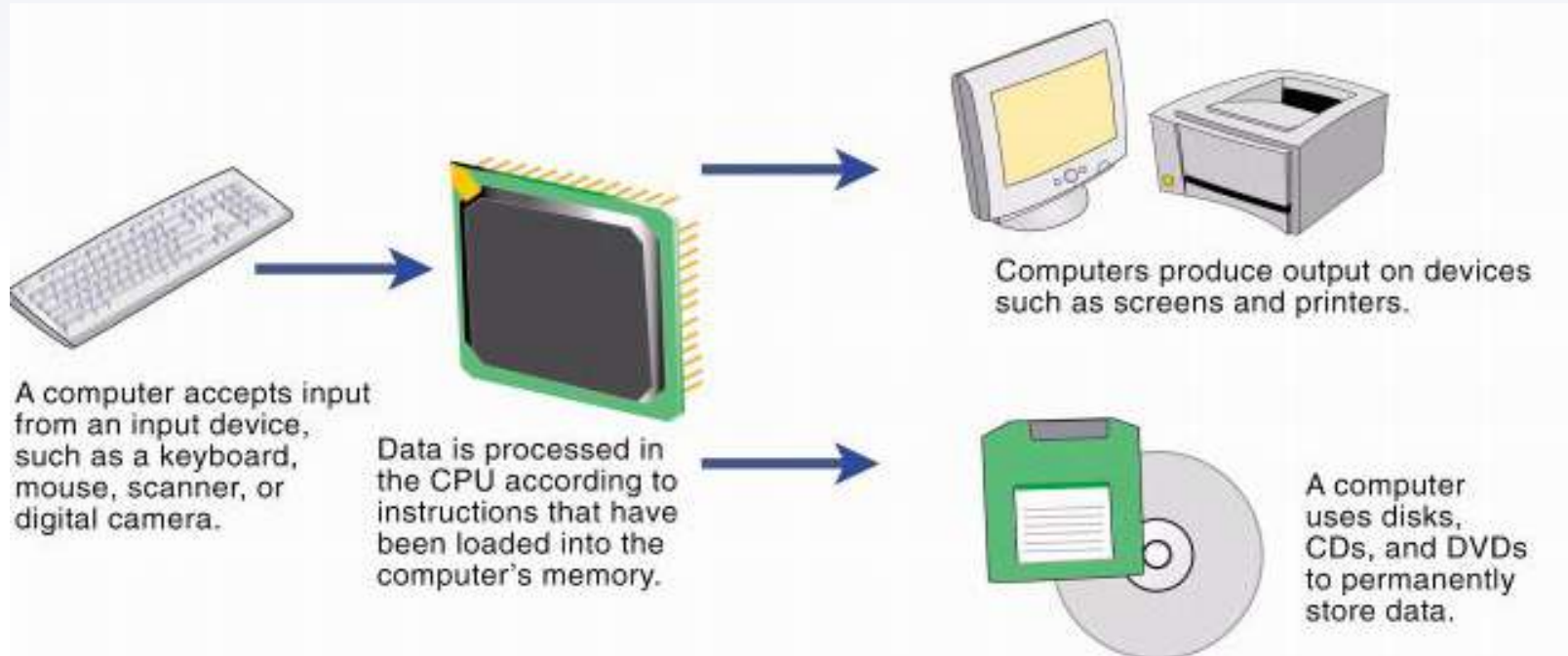


WHAT IS COMPUTER

- ***Memory*** is an area of a computer that holds data that is waiting to be processed, stored, or output
- ***Storage*** is the area where data can be left on a permanent basis
- **Computer *output*** is the result produced by the computer
- **An output device** displays, prints or transmits the results of processing



PROCESSING OF A COMPUTER



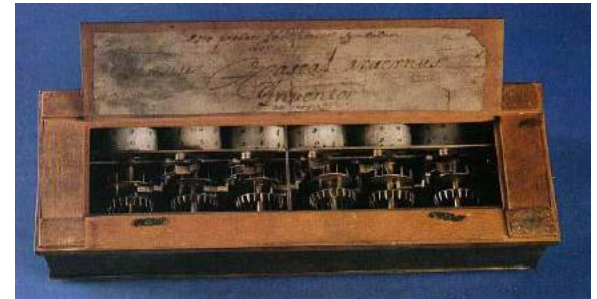
BEFORE INVENTION OF A COMPUTER

- Before the 1500s, in Europe, calculations were made with an **abacus**
 - ❓ Invented around 500BC, available in many cultures (China, Mesopotamia, Japan, Greece, Rome, etc.)



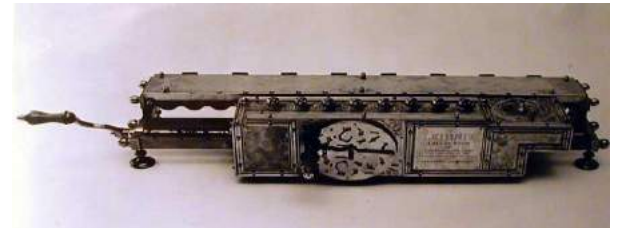
BEFORE INVENTION OF A COMPUTER

In 1642, **Blaise Pascal** (French mathematician, physicist, philosopher) invented a mechanical calculator called the **Pascaline**



BEFORE INVENTION OF A COMPUTER

- In 1671, **Gottfried von Leibniz** (German mathematician, philosopher) extended the Pascaline to do multiplications, divisions, square roots: the **Stepped Reckoner**

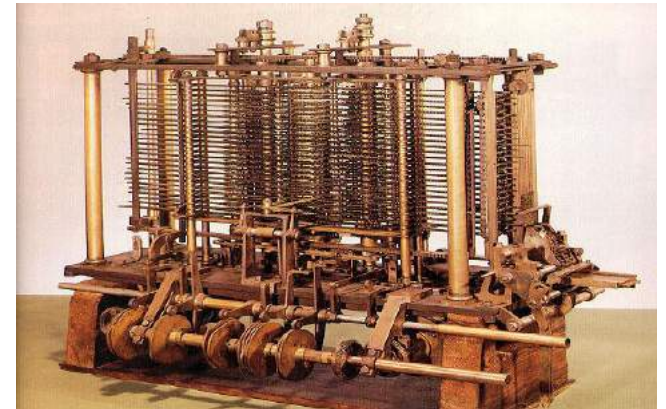
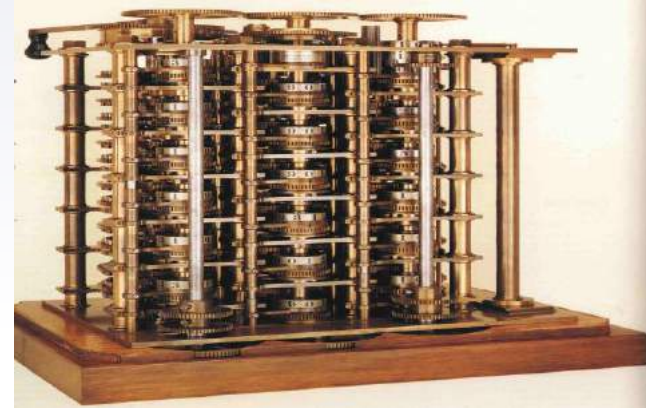


DRAWBACK

None of these machines had memory, and they required human intervention at each step

INVENTION OF COMPUTER

- In 1822 **Charles Babbage** (English mathematician, philosopher), sometimes called the “father of computing” built the **Difference Engine**
- Machine designed to automate the computation (tabulation) of polynomial functions (which are known to be good approximations of many useful functions)
 - Based on the “method of finite difference”
 - Implements some storage
- In 1833 Babbage designed the **Analytical Engine**, but he died before he could build it
 - It was built after his death, powered by steam



GENERATION OF COMPUTER

- First Generation (1946-59)
- Second Generation(1957-64)
- Third Generation(1965-70)
- Fourth Generation(1970-90)
- Fifth Generation(1990 till date)

COMPONENT USED

Generation 0: Mechanical Calculators

Generation 1: Vacuum Tube Computers

Generation 2: Transistor Computers

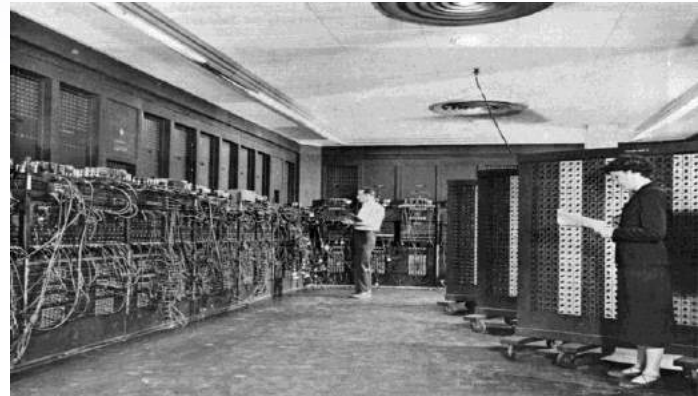
Generation 3: Integrated Circuits

Generation 4: Microprocessors

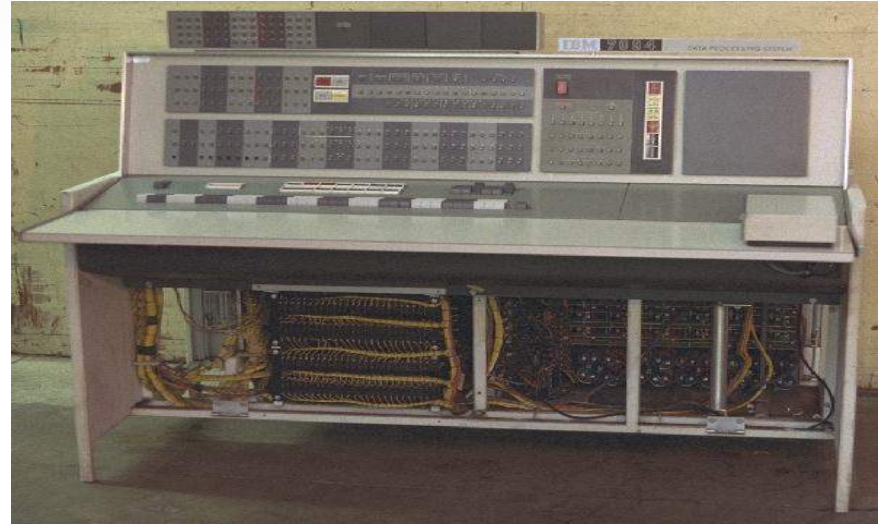


Generation 1 : ENIAC

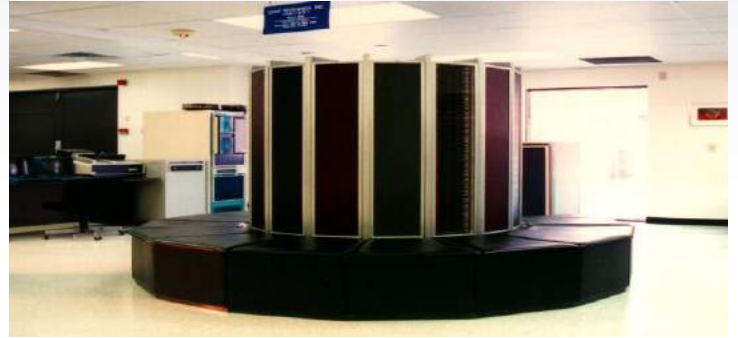
The ENIAC (Electronic Numerical Integrator and Computer) was unveiled in 1946: the first all-electronic, general-purpose digital computer



Generation 2: IBM7094



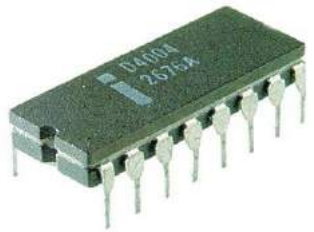
Generation 3: Integrated Circuits



Seymour Cray created the Cray Research Corporation
Cray-1: \$8.8 million, 160 million instructions per
seconds and 8 Mbytes of memory



Generation 4: VLSI



Microprocessors

Improvements to IC technology made it possible to integrate more and more transistors in a single chip

SSI (Small Scale Integration): 10-100

MSI (Medium Scale Integration): 100-1,000

LSI (Large Scale Integration): 1,000-10,000

VLSI (Very Large Scale Integration): >10,000



Generation 5

