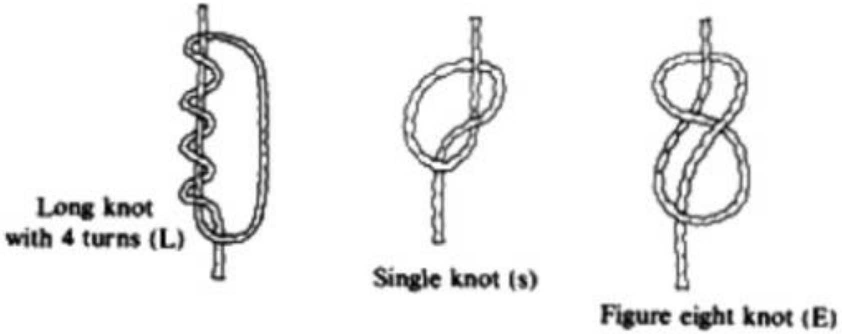
The Discipline of Computing

- Computing milestones and machine evolution
- Generations of computing
- Evolution of computing

Computing milestones and machine evolution

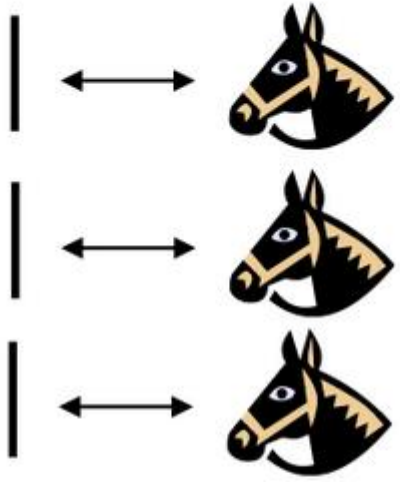
- In ancient times people used stones for counting.

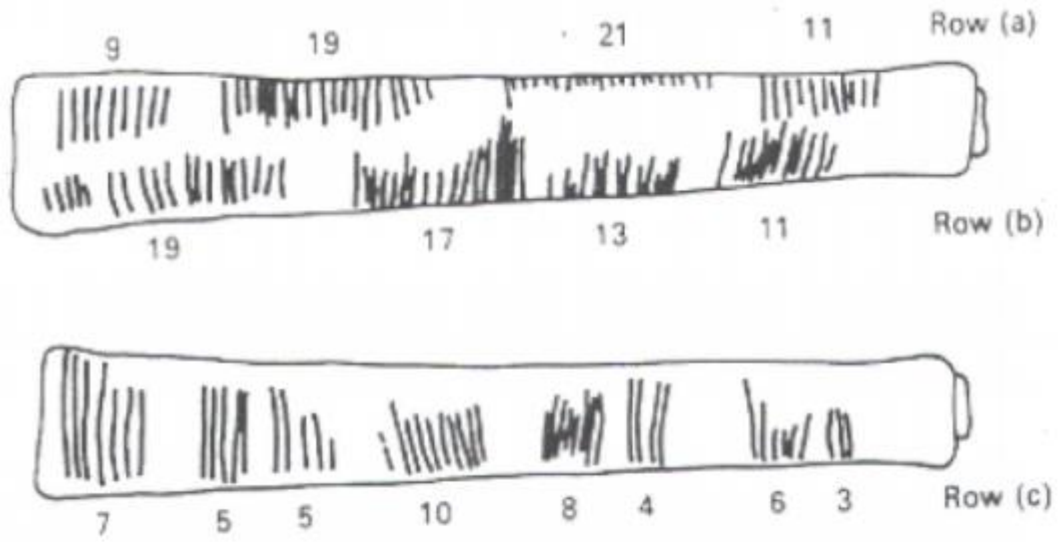
- The maid scratches on wall or tied knots in ropes to record information



Counting and Evolution of the Positional Number System

- Stick representation: Each stick represent one animal or object ,
  Each number has a weight





Different Counting methods & Number systems

→ Egyptian Number system

→ Sumerian/Babylonian Number system

→ Chinese Number system

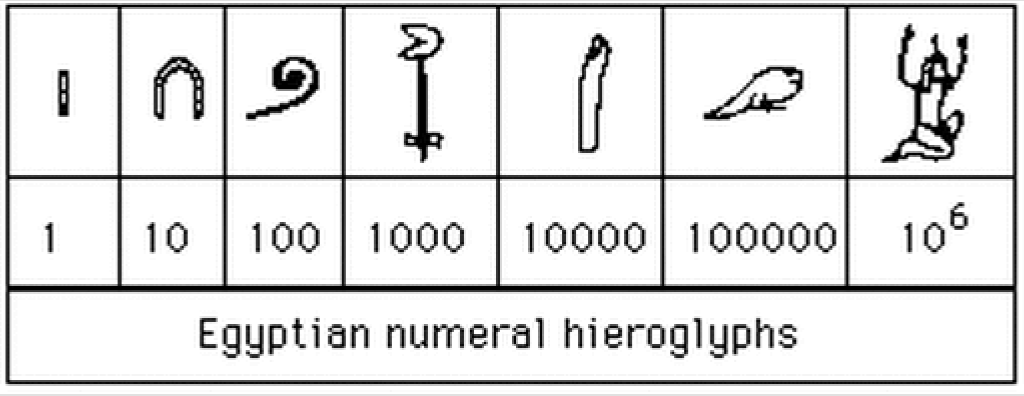
→ Greek / Ionian Number system

→ Romans Number system

→ Mayans Number system

→ Hindu-Arabic Numeral system

- Egyptian Number system
  - 3000BC
  - Radix / Base : 10
  - Right to left



1	10	100	1000	10000	100000	10^6

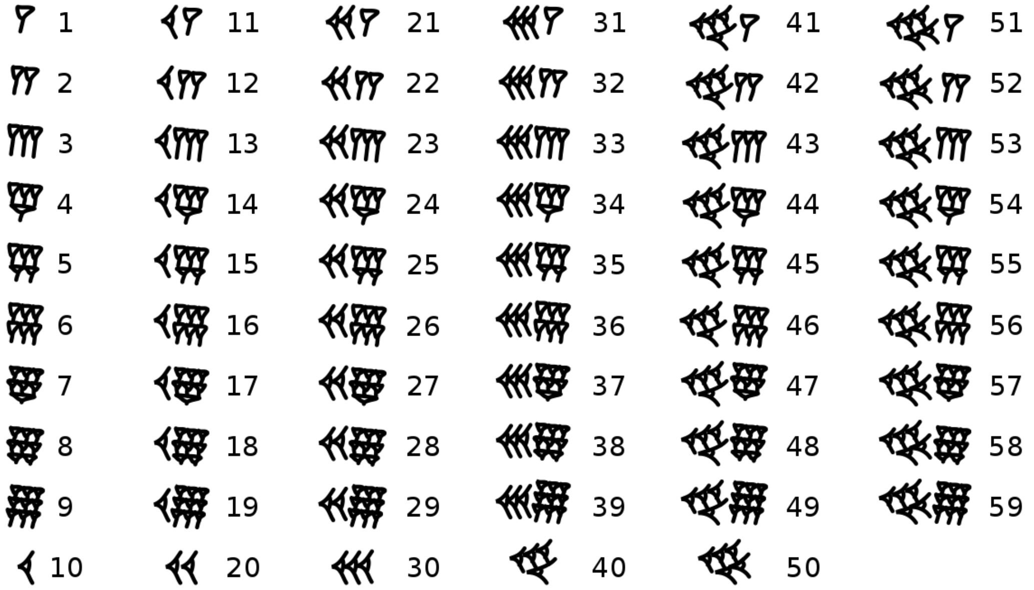
Egyptian numeral hieroglyphs

(base / radix : 2)

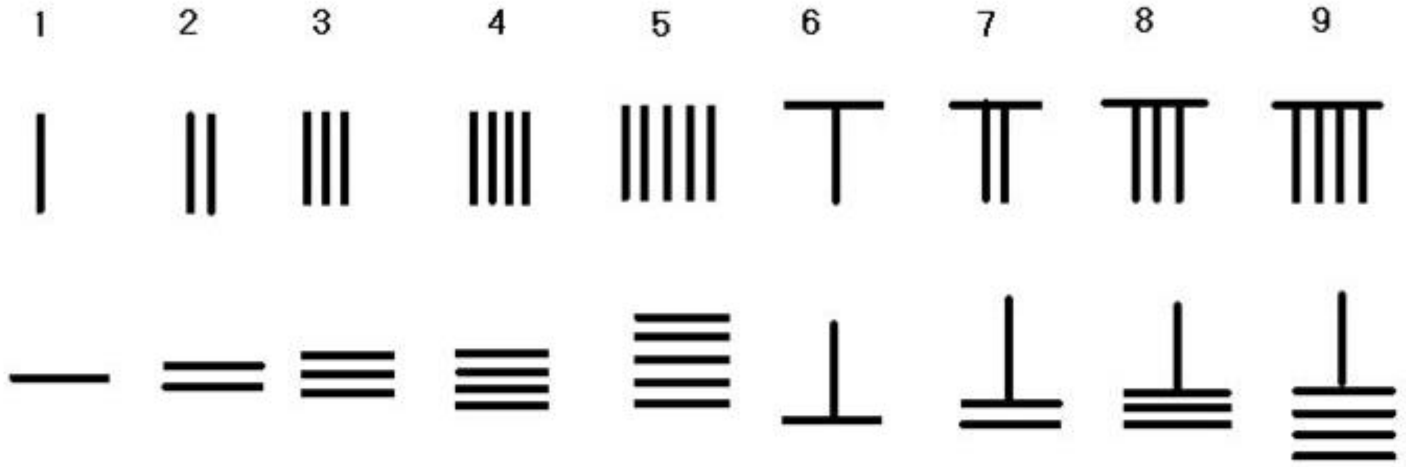
- $(1101.11)_2$ ← Binary Number system

Sumerian / Babylonian Number system

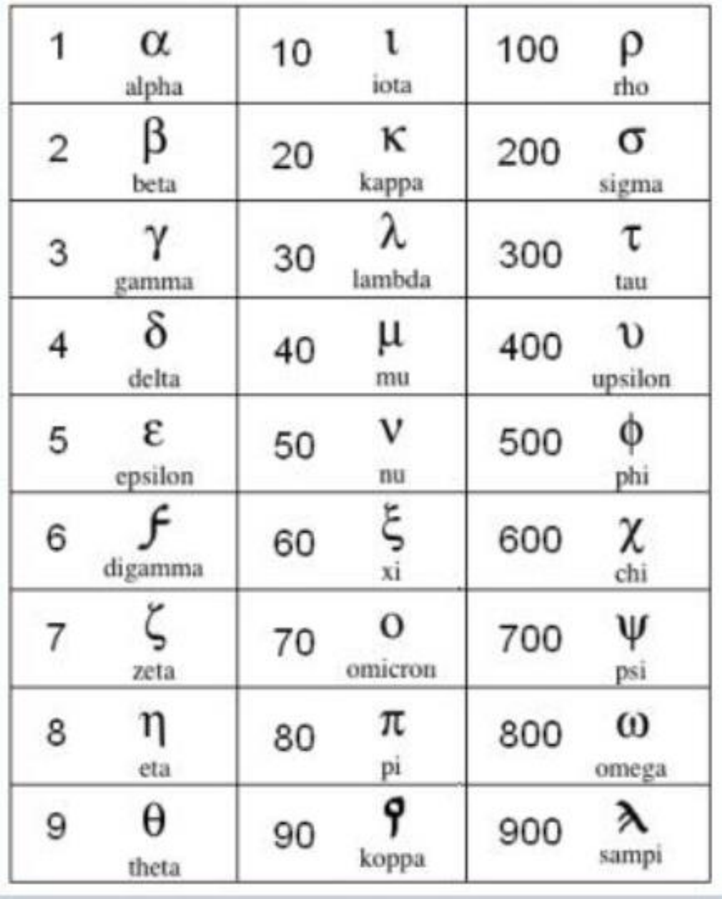
- 2000BC
- Radix / Base : 60
- Left to right
- Also called sexagesimal number



- Chinese Number system
  - 2500BC
  - 0 to 9
  - Represent number : Bamboo rods
  - Left to right



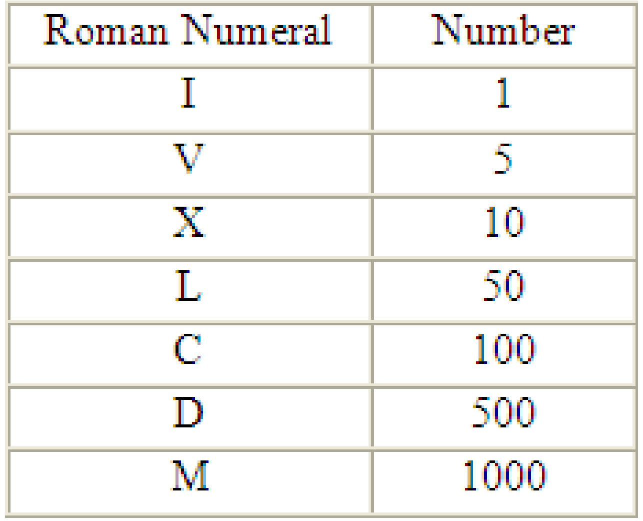
- Greek / Ionian Number system
  - 500BC
  - Base / radix : 10
  - Decimal Number system
  - Left to right



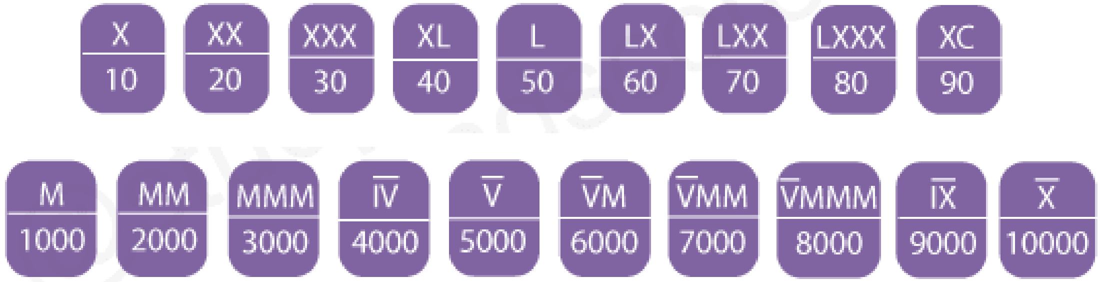
1	α alpha	10	ι iota	100	ρ rho
2	β beta	20	κ kappa	200	σ sigma
3	γ gamma	30	λ lambda	300	τ tau
4	δ delta	40	μ mu	400	υ upsilon
5	ε epsilon	50	ν nu	500	ϕ phi
6	ϝ digamma	60	ξ xi	600	χ chi
7	ζ zeta	70	ο omicron	700	ψ psi
8	η eta	80	π pi	800	ω omega
9	θ theta	90	ϙ koppa	900	ϡ sampi

Romans Number system

- 800BC - 900BC
- Base / Radix : 10
- Use 7 letter to represent numbers
- I V X L C D M
- Left to right

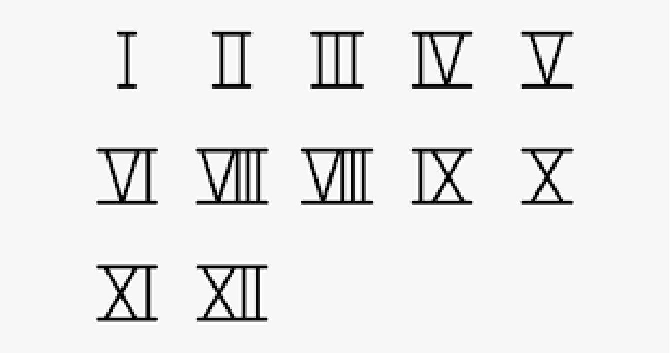


Roman Numeral	Number
I	1
V	5
X	10
L	50
C	100
D	500
M	1000



X	XX	XXX	XL	L	LX	LXX	LXXX	XC
10	20	30	40	50	60	70	80	90

M	MM	MMM	$\overline{\text{IV}}$	$\overline{\text{V}}$	$\overline{\text{V}}$M	$\overline{\text{V}}$MM	$\overline{\text{V}}$MMM	$\overline{\text{IX}}$	$\overline{\text{X}}$
1000	2000	3000	4000	5000	6000	7000	8000	9000	10000

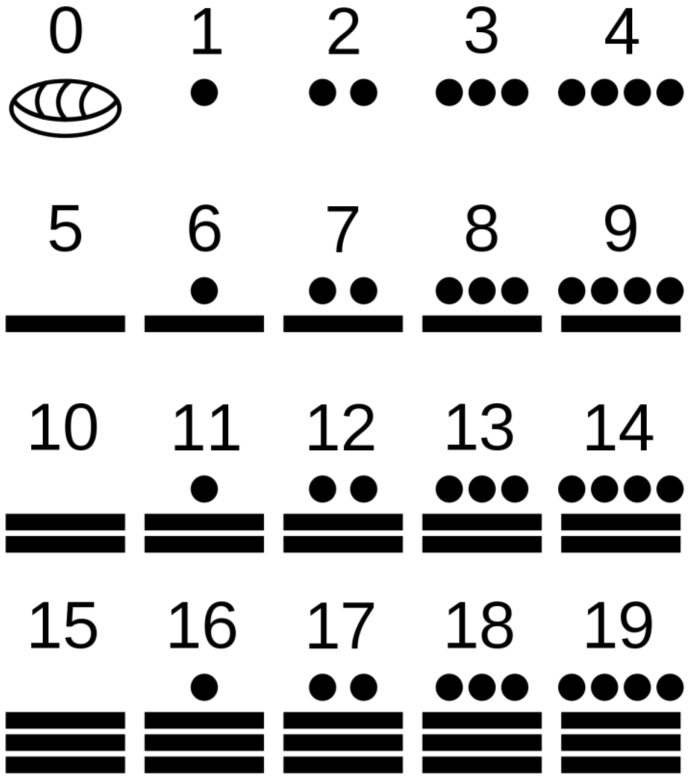


I II III IV V

VI VIII VIII IX X

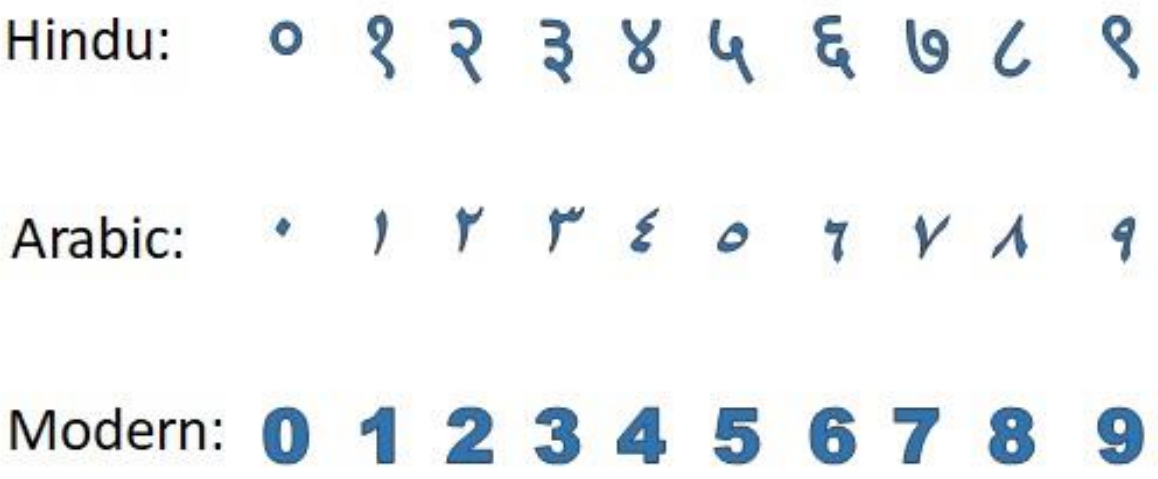
XI XII

- Mayans Number system
  - 400BC – 150AD (Maya civilization)
  - Base / Radix : 20
  - Made up of 3 symbols :
  - 0 : shell shape, 1 : dot, 5 : a bar



Hindu-Arabic Numeral system

- 6th or 7th century
- Base / radix : 10
- Decimal Number system (0 to 9)
- It is a positional number system
- Originated in India
- Widely used today



Hindu:	०	१	२	३	४	५	६	७	८	९
Arabic:	٠	١	٢	٣	٤	٥	٦	٧	٨	٩
Modern:	0	1	2	3	4	5	6	7	8	9

Hindu-Arabic

Evolution of the Computing Machine

→ Abacus

→ Napier’s bones

→ Pascaline

→ Leibniz’s calculator

→ Jacquard’s loom

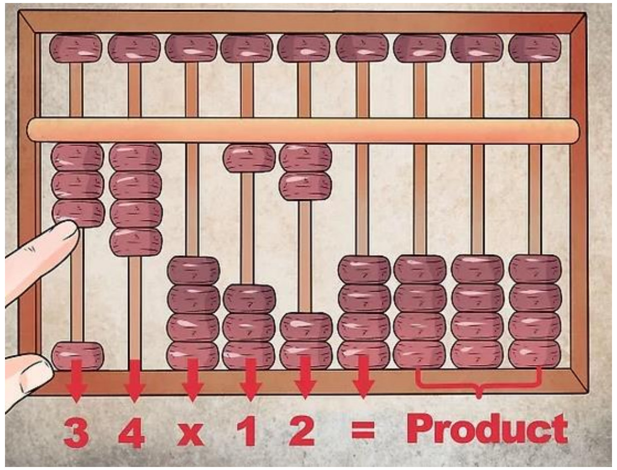
→ Difference engine

→ Analytical engine

→ Hollerith’s machine

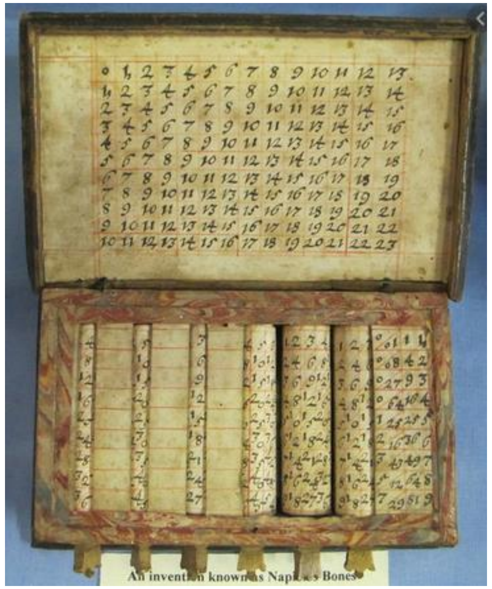
→ Mark -i

- Abacus
  - 3000BC
  - Manual Calculator
  - First computer (Basic Arithmetic Calculations)
  - Abacus meaning : calculating board / also called courting frame
  - Discovered by : Mesopotamians
  - Works on the basis of place value system

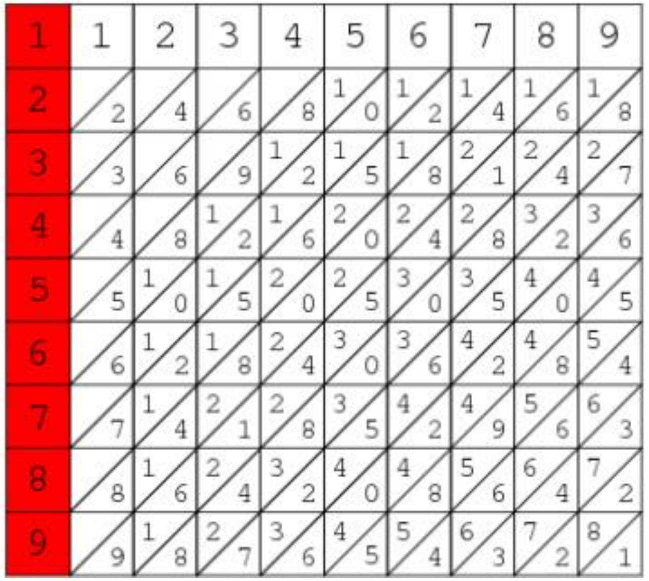


Napier’s bones

- 1617 AD
- Manual Calculator
- Created by : John Napier
- Calculation : Division, Square root, Multiplication, Addition, Subtraction

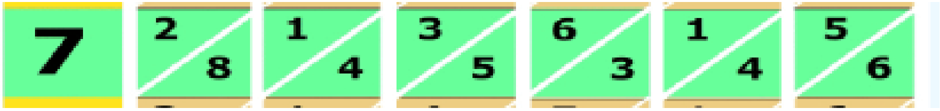


An invention known as Napier's Bones



1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	1/0	1/2	1/4	1/6	1/8
3	3	6	9	1/2	1/5	1/8	2/1	2/4	2/7
4	4	8	1/2	1/6	2/0	2/4	2/8	3/2	3/6
5	5	1/0	1/5	2/0	2/5	3/0	3/5	4/0	4/5
6	6	1/2	1/8	2/4	3/0	3/6	4/2	4/8	5/4
7	7	1/4	2/1	2/8	3/5	4/2	4/9	5/6	6/3
8	8	1/6	2/4	3/2	4/0	4/8	5/6	6/4	7/2
9	9	1/8	2/7	3/6	4/5	5/4	6/3	7/2	8/1

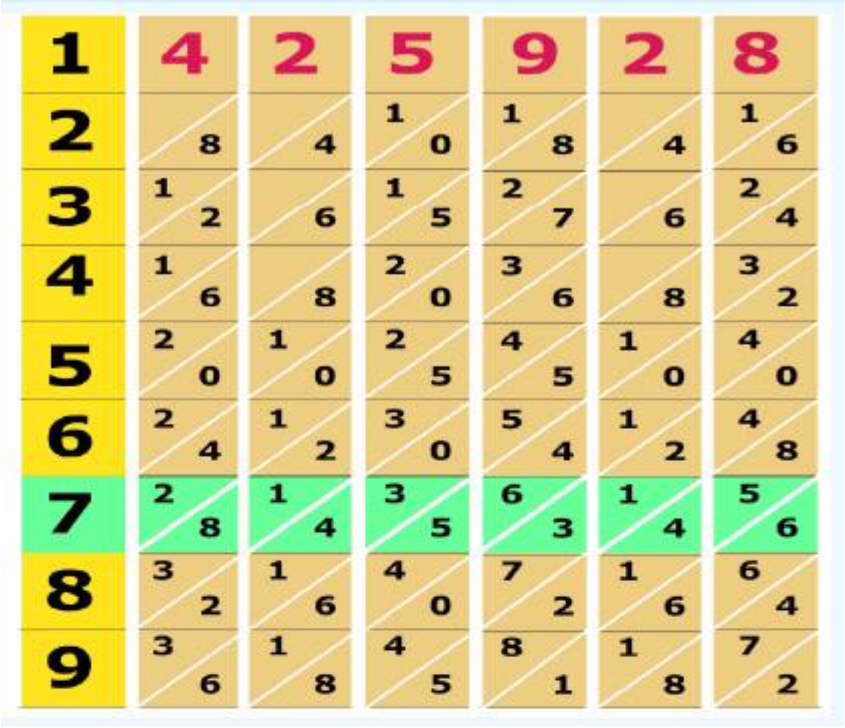
7 x 425928 = ?



7	2/8	1/4	3/5	6/3	1/4	5/6

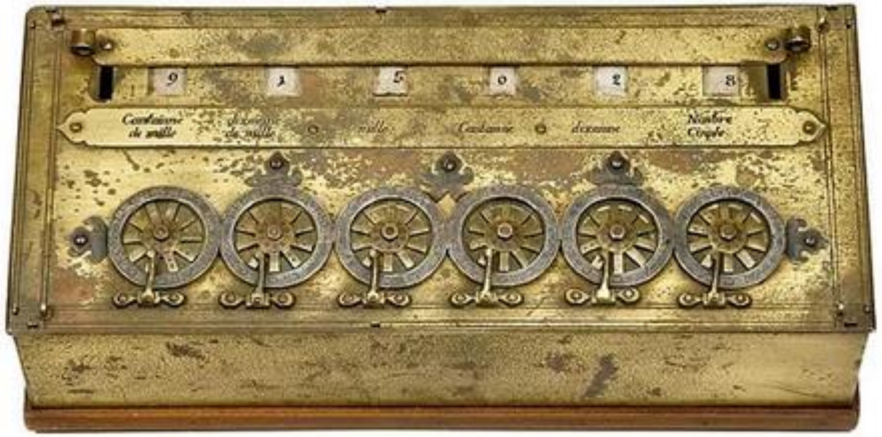
2(8+1) (4+3) (5+6) (3+1) (4+5)6

= 2981496

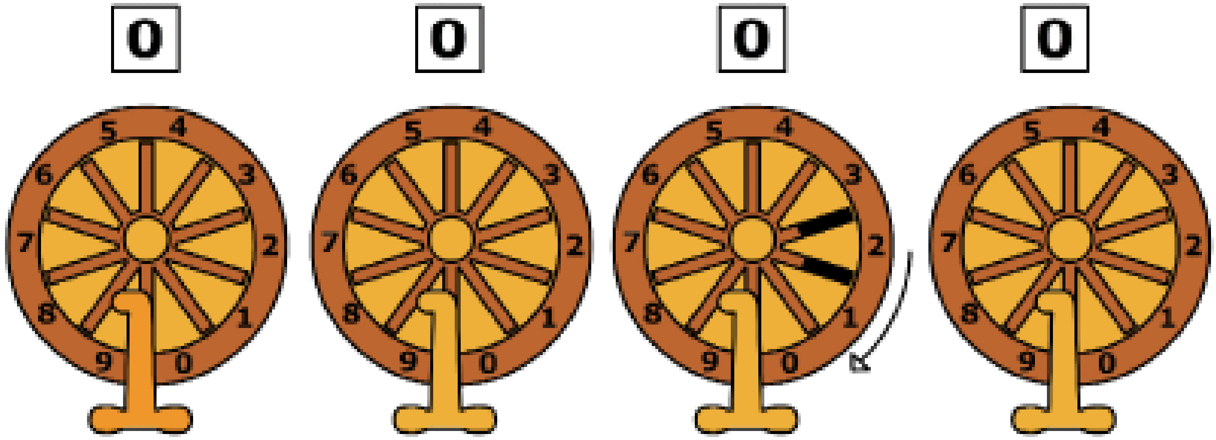


1	4	2	5	9	2	8
2	8	4	1/0	1/8	4	1/6
3	1/2	6	1/5	2/7	6	2/4
4	1/6	8	2/0	3/6	8	3/2
5	2/0	1/0	2/5	4/5	1/0	4/0
6	2/4	1/2	3/0	5/4	1/2	4/8
7	2/8	1/4	3/5	6/3	1/4	5/6
8	3/2	1/6	4/0	7/2	1/6	6/4
9	3/6	1/8	4/5	8/1	1/8	7/2

- Pascaline
  - 1642 (17th century)
  - Mechanical Calculator
  - Use : Adding and subtracting two numbers directly,
    Multiplication and division through repeated addition or subtraction
  - It have a series of wheels, gear and cylinders



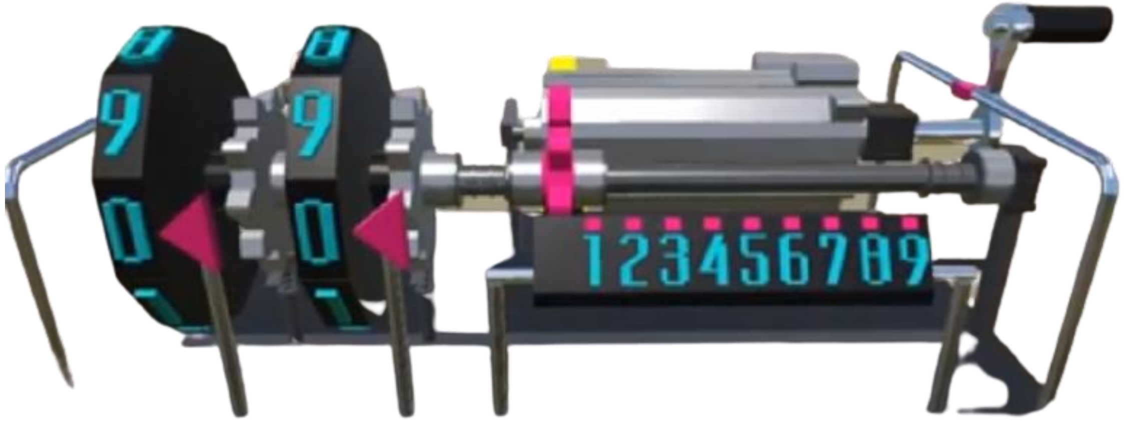
20 + 81 = ?



https://www.edumedia-ciences.com/en/media/333-pascaline

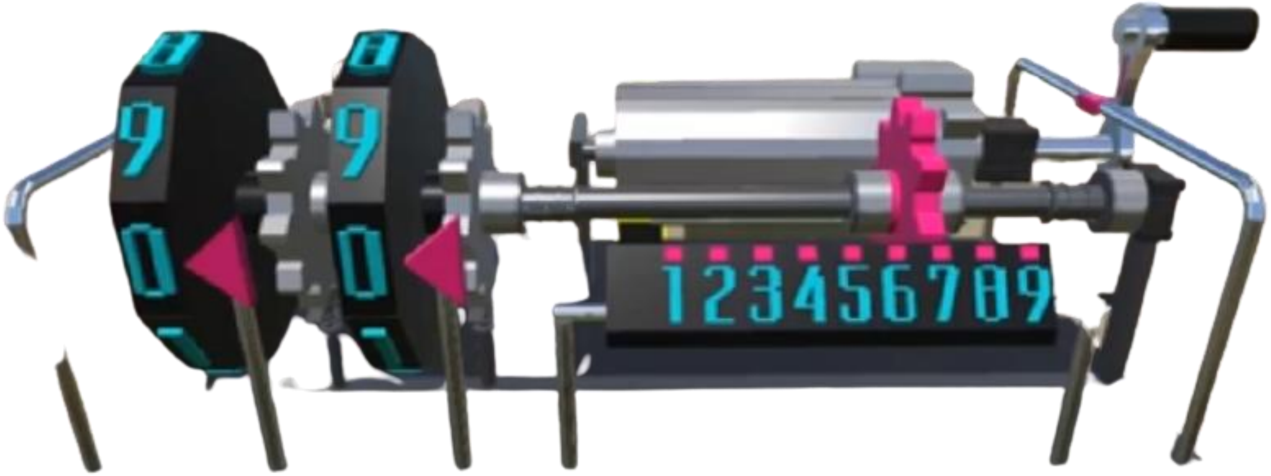
- Leibniz’s calculator
  - 1673
  - Mechanical Calculator called Step Reckoner
  - Extended pascals idea and extended the capability
  - Perform multiplication and division as well

7 x 5 = ?



0 0 ⬆

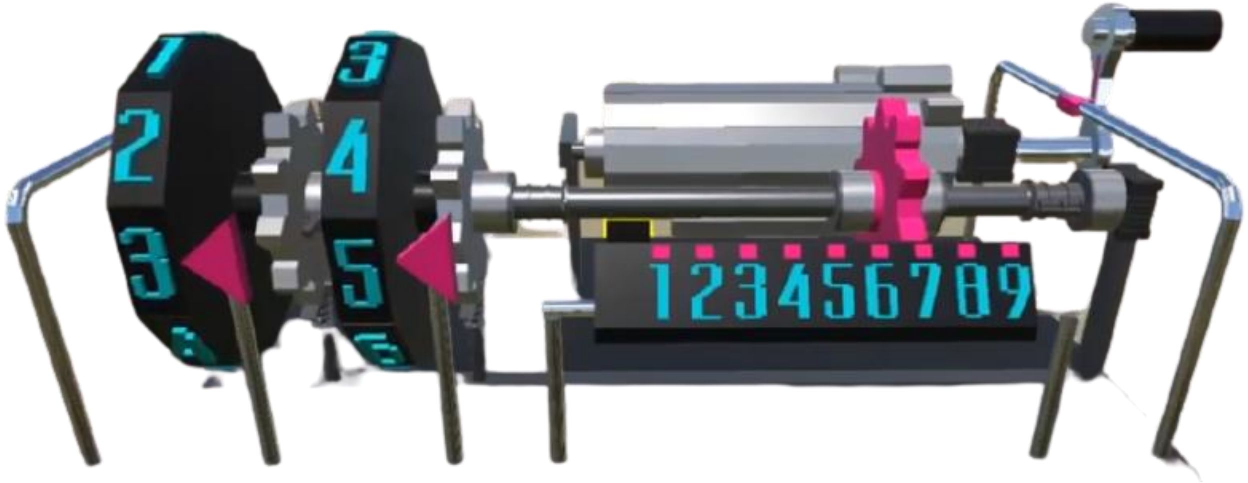
Rotate 5 times



0 0

⬆

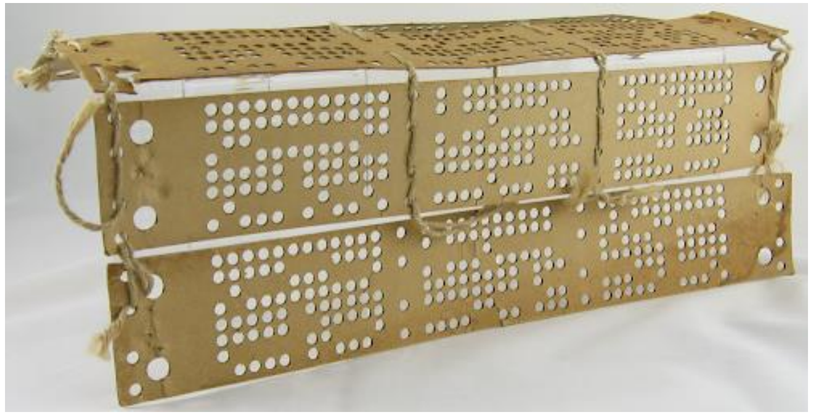
Set Gear on 7

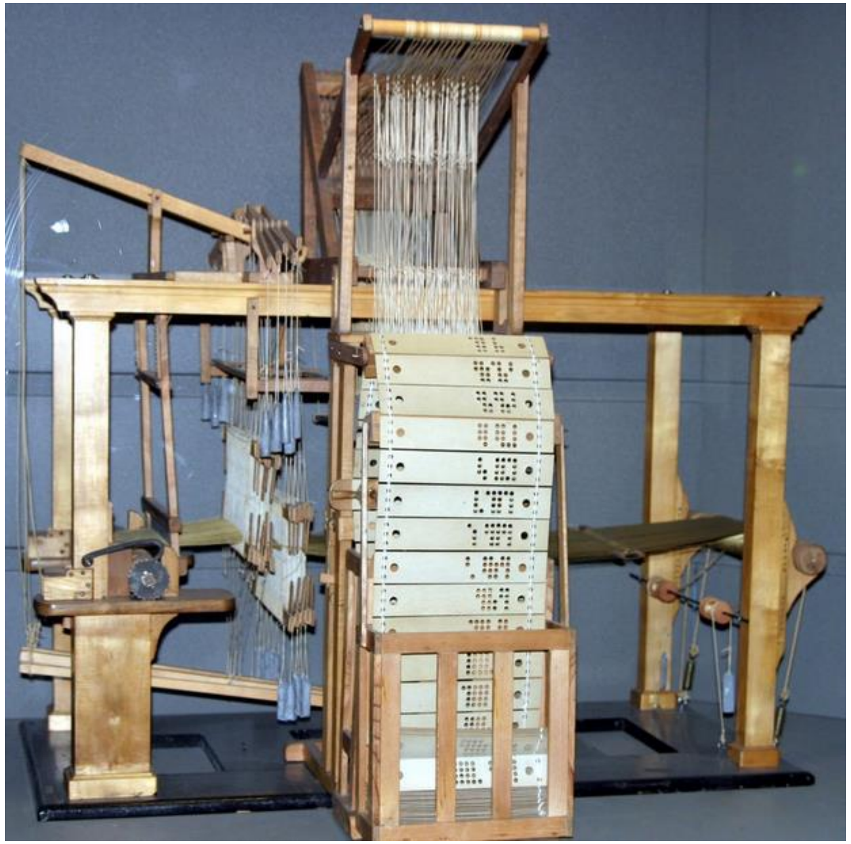


3 5

After 5 time rotation

- Jacquard’s loom
  - 1801
  - Simplify the process of manufacturing textiles with complex patterns
  - Loop controlled by punched cards (holes)
  - It allowed to store Patterson card (use to create the same product again)
  - This store information triggered the computer revolution
  - Punch card adopted by Charles Babbage





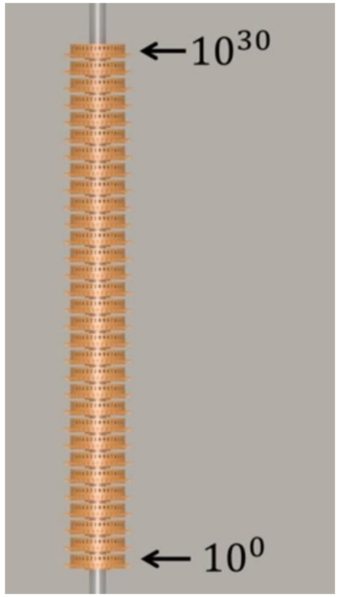
https://www.youtube.com/watch?v=awGjOGo_Mis

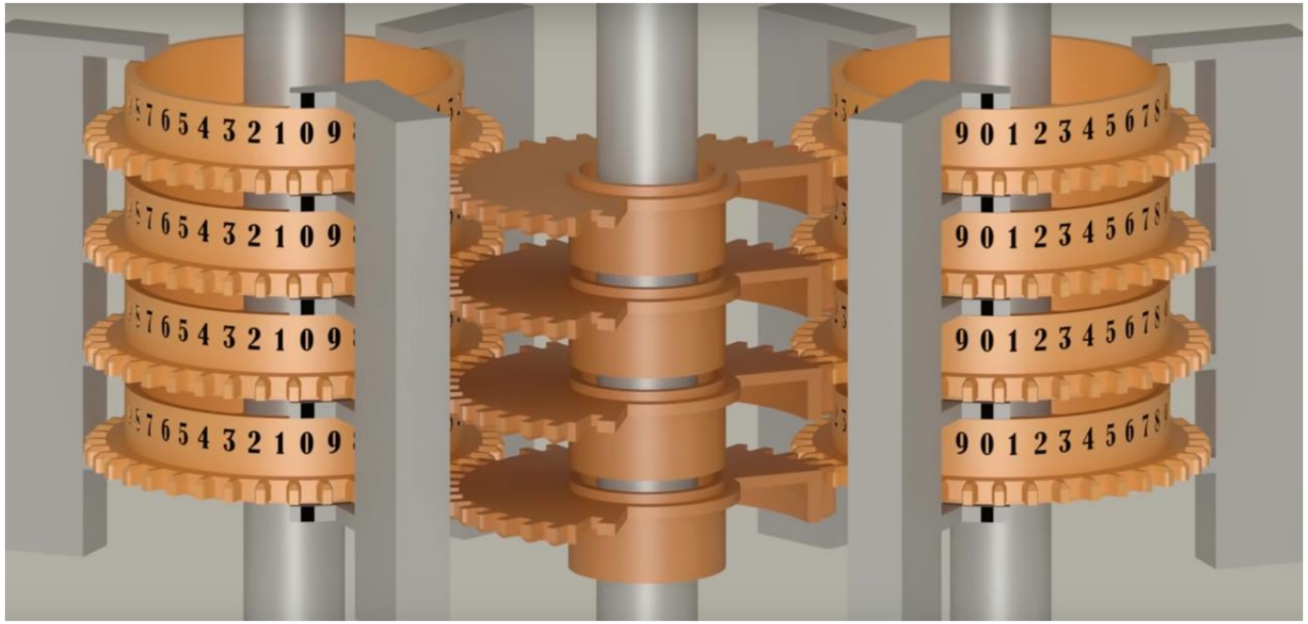
- Difference engine
  - 1822
  - Automatic mechanical calculator
  - First step towards the creation of computer
  - Created by : Charles Babbage
  - Use : to compile mathematical table

    (designed to tabulate polynomial functions)

7 6 5 4 3 2 1 0 9 8 7 6 5 4

0 t0 9 X 4 parts

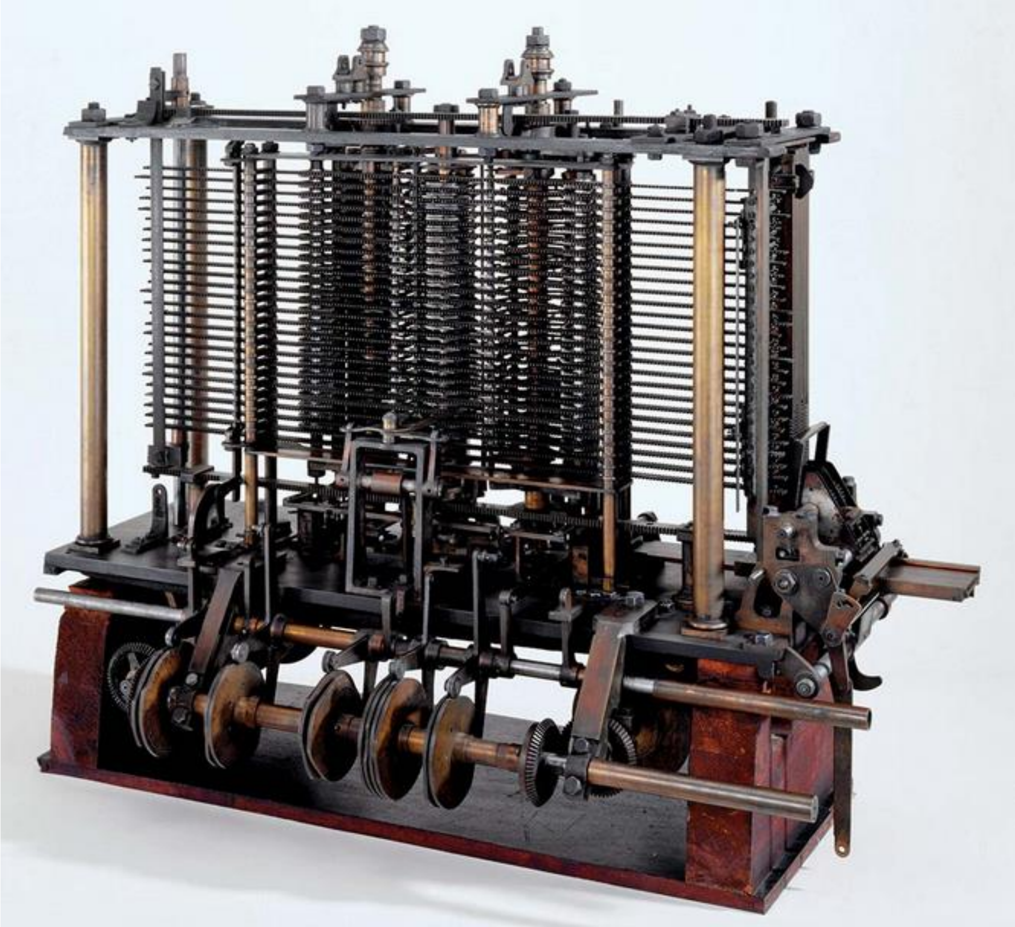




https://www.youtube.com/watch?v=be1EM3gQkAY

- Analytical engine
  - 1837
  - Automatic mechanical calculator
  - the concept for the first general mechanical computer
  - Created by : Charles Babbage
  - Design feature : memory (store number and intermediate result)
  - Separate mill(processor or ALU)
  - Input / output device (program written in the form of punching card)
  - Agusta Ada king (**Ada Lovelace :** First programmer in the world)

Ada Lovelace

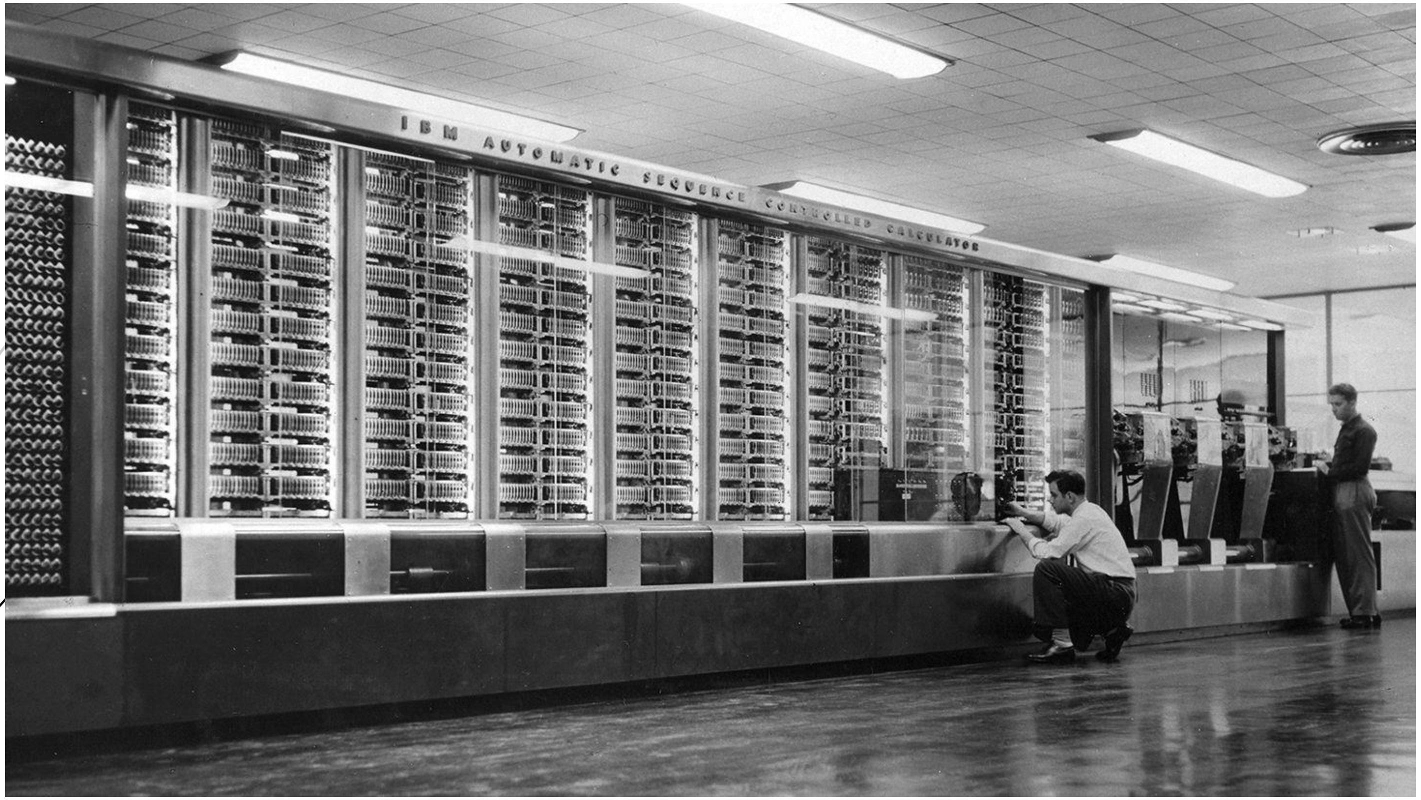


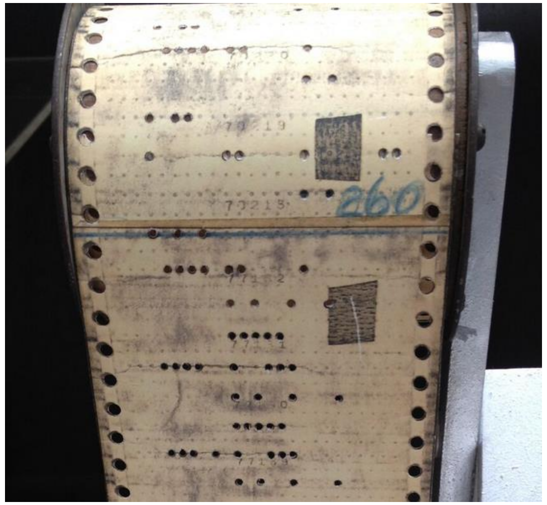
- Hollerith's machine
  - 1887
  - Electro mechanical Punched card machine also called Tabulating machine
  - Created by : Herman Hollerith
  - Machine used : input punching card creation,
    to read result of a card, new programming

https://www.youtube.com/watch?v=17On5ltcrBA

- Mark - i
  - 1944
  - Large Automatic electromechanical computer
  - Created by : Howard Aiken
  - Also called Aiken’s machine
  - Preform : all the arithmetic operation, logarithm and trigonometric functions
  - Inputs / output : paper tape readers, card readers, card punch, typewriters

https://www.youtube.com/watch?v=17On5ItcrBA





Mark – i / Aiken’s machine

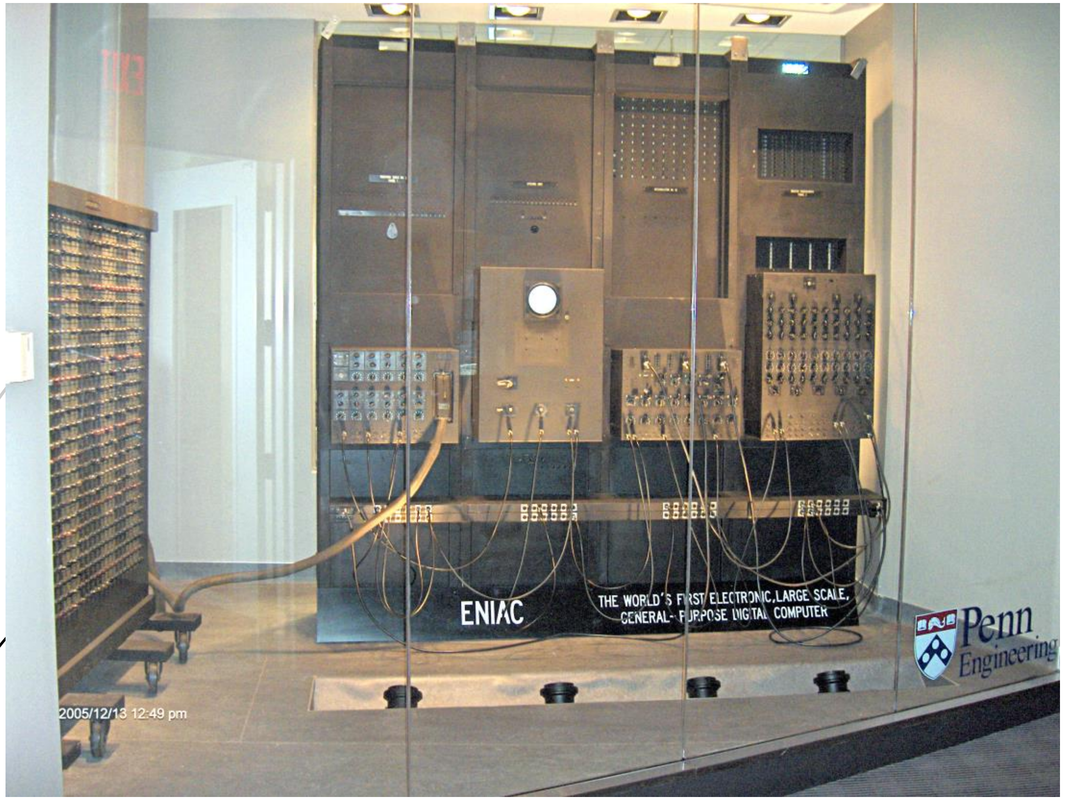
Generations of Computes

Starting from 16th century

- First Generation Computers
  - Von Neumann Architecture
- Second Generation Computers
- Third Generation Computers
- Fourth Generation Computers (Present)
- Fifth Generation Computers (Present & Future)

→ First Generation Computers

- 1940 - 1956
- Vacuum Tubes
- First General Purpose Programming Electronic Computer
- Electronic Numerical Integrator and Calculator (ENIAC) → store Program
- Electronic Discrete Variable Automatic Computer (EDVAC) → store program + data
- It use Von Neumann design
- First commercially successful computer
- Universal Automatic Computer (UNIVAC)


ENIAC
THE WORLD'S FIRST ELECTRONIC, LARGE SCALE, GENERAL-PURPOSE DIGITAL COMPUTER
Penn Engineering
2005/12/13 12:49 pm

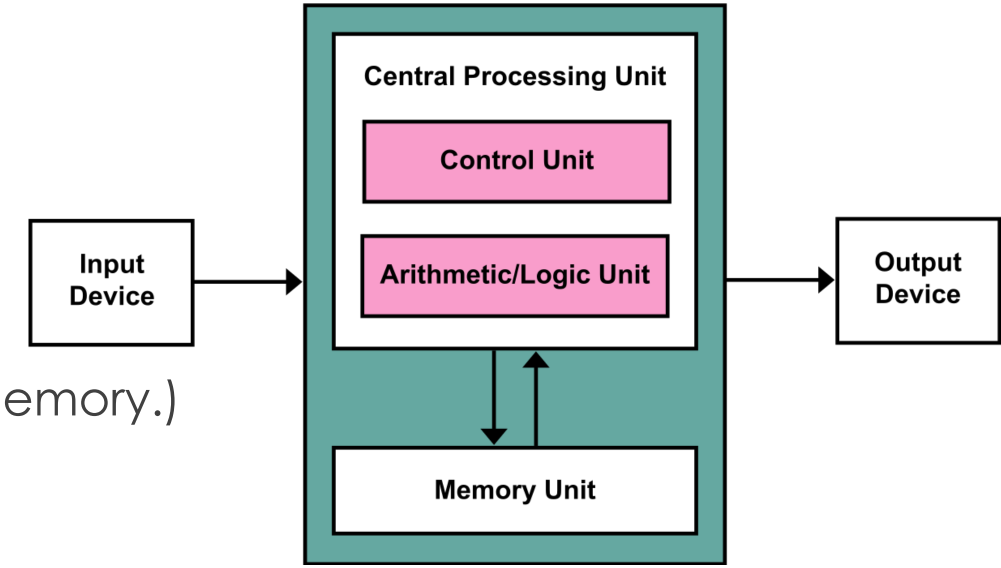
ENIAC

EDVAC

UNIVAC

→ Von Neumann design

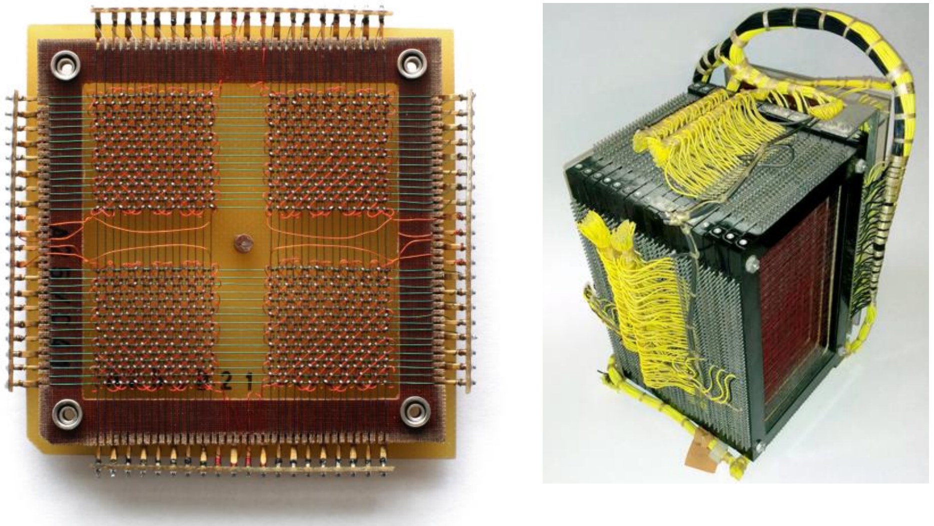
- CPU contain ALU and CU
- Input / output unit
- Memory for store data and program
- (Data and Program are store in same memory.)



- **Harvard architecture** : data memory (RAM) and program memory (ROM) are separately.
- **Von-Neumann architecture** : data and program are store in same memory.

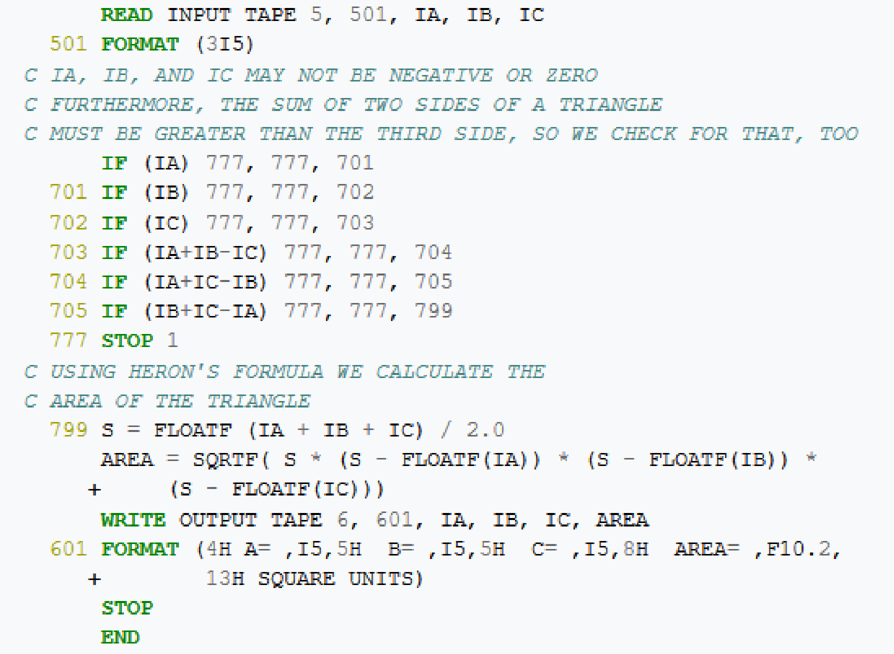
→Second Generation Computers

- 1956 – 1963
- Transistor
- It reduce the size and increase the performanc
- Reduce cost, reduce heat, reduce use of elect
- Magnetic core memory and magnetic disk
- Programming language developed (HLL : FORTRAN and COBOL introduced)
- Eg : IBM 1041 and IBM 1620



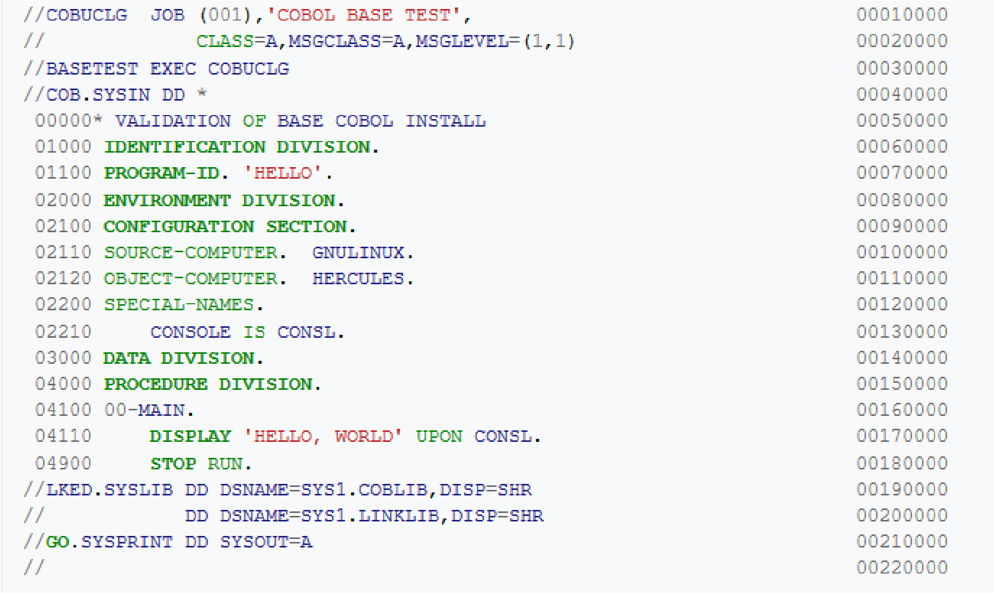
Magnetic core memory

FORTRAN



```
       READ INPUT TAPE 5, 501, IA, IB, IC
   501 FORMAT (3I5)
C IA, IB, AND IC MAY NOT BE NEGATIVE OR ZERO
C FURTHERMORE, THE SUM OF TWO SIDES OF A TRIANGLE
C MUST BE GREATER THAN THE THIRD SIDE, SO WE CHECK FOR THAT, TOO
       IF (IA) 777, 777, 701
   701 IF (IB) 777, 777, 702
   702 IF (IC) 777, 777, 703
   703 IF (IA+IB-IC) 777, 777, 704
   704 IF (IA+IC-IB) 777, 777, 705
   705 IF (IB+IC-IA) 777, 777, 799
   777 STOP 1
C USING HERON'S FORMULA WE CALCULATE THE
C AREA OF THE TRIANGLE
   799 S = FLOATF (IA + IB + IC) / 2.0
       AREA = SQRTF( S * (S - FLOATF(IA)) * (S - FLOATF(IB)) *
      +     (S - FLOATF(IC)))
       WRITE OUTPUT TAPE 6, 601, IA, IB, IC, AREA
   601 FORMAT (4H A= ,I5,5H  B= ,I5,5H  C= ,I5,8H  AREA= ,F10.2,
      +        13H SQUARE UNITS)
       STOP
       END
```

COBOL



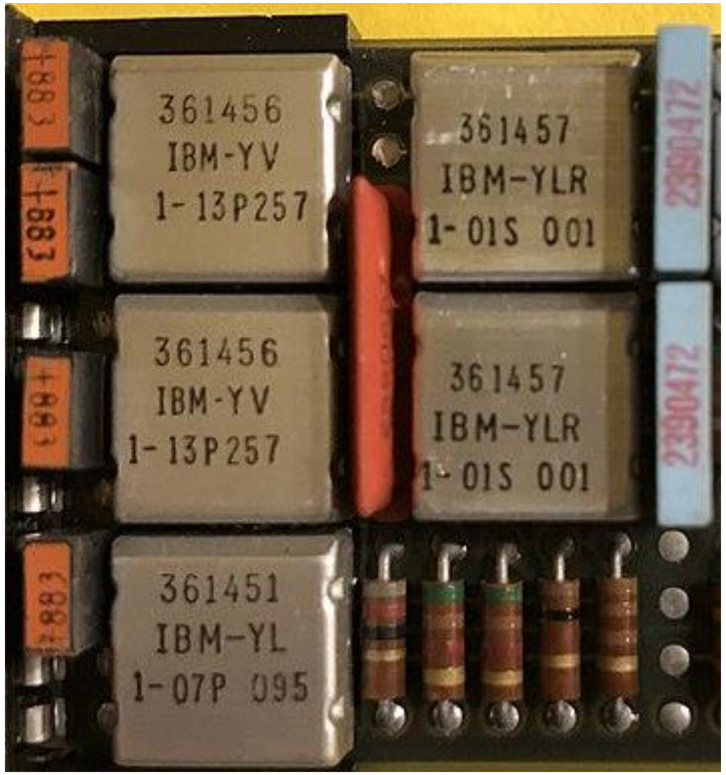
```
//COBUCLG  JOB (001),'COBOL BASE TEST',                                 00010000
//             CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)                        00020000
//BASETEST EXEC COBUCLG                                                 00030000
//COB.SYSIN DD *                                                        00040000
 00000* VALIDATION OF BASE COBOL INSTALL                                00050000
 01000 IDENTIFICATION DIVISION.                                         00060000
 01100 PROGRAM-ID. 'HELLO'.                                             00070000
 02000 ENVIRONMENT DIVISION.                                            00080000
 02100 CONFIGURATION SECTION.                                           00090000
 02110 SOURCE-COMPUTER.  GNULINUX.                                      00100000
 02120 OBJECT-COMPUTER.  HERCULES.                                      00110000
 02200 SPECIAL-NAMES.                                                   00120000
 02210     CONSOLE IS CONSL.                                            00130000
 03000 DATA DIVISION.                                                   00140000
 04000 PROCEDURE DIVISION.                                              00150000
 04100 00-MAIN.                                                         00160000
 04110     DISPLAY 'HELLO, WORLD' UPON CONSL.                           00170000
 04900     STOP RUN.                                                    00180000
//LKED.SYSLIB DD DSNAME=SYS1.COBLIB,DISP=SHR                            00190000
//            DD DSNAME=SYS1.LINKLIB,DISP=SHR                           00200000
//GO.SYSPRINT DD SYSOUT=A                                               00210000
//                                                                      00220000
```

IBM 1041

IBM 1620

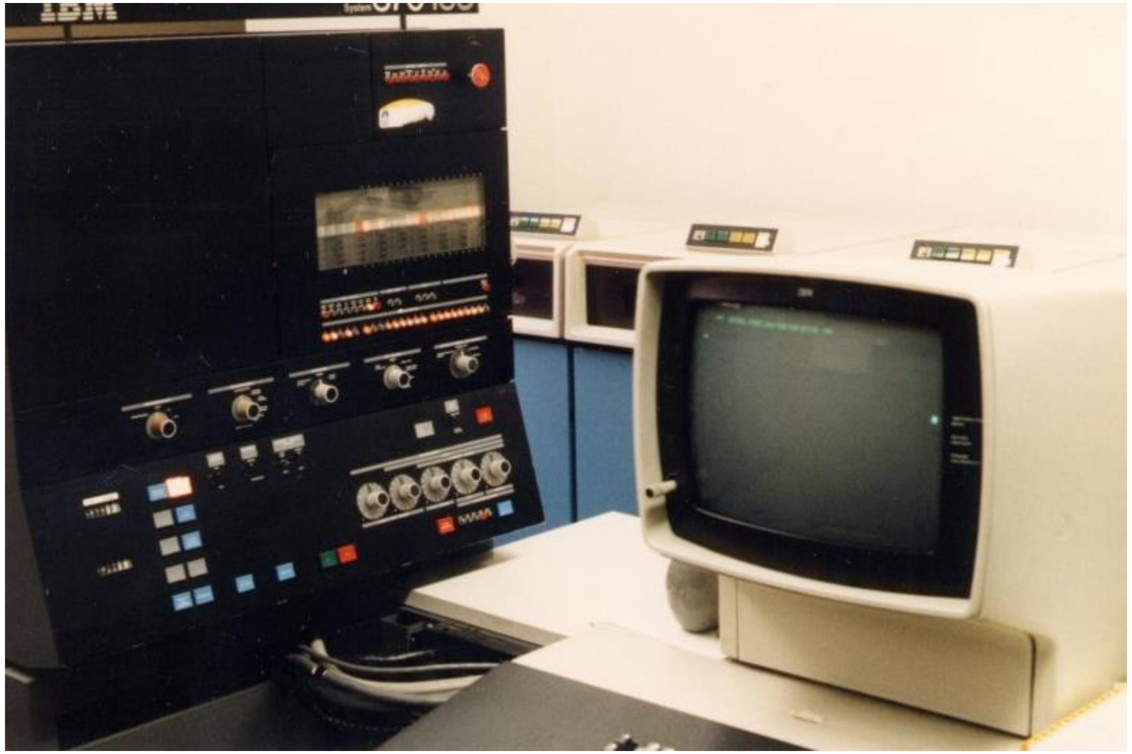
→Third Generation Computers

- 1964 – 1971
- IC (Integrated Circuits)
- Again reduce the size
- increase the performance speed and efficiency
- Monitor & Keyboard
- HLL : BASIC)
- Eg : IBM 360 and IBM 370

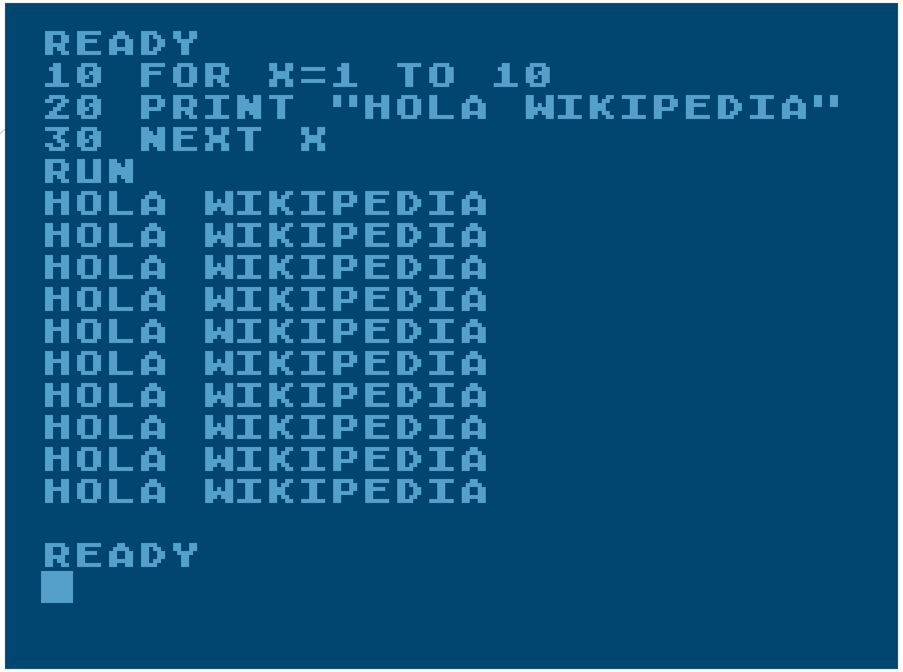


IC

IBM 360



IBM 370



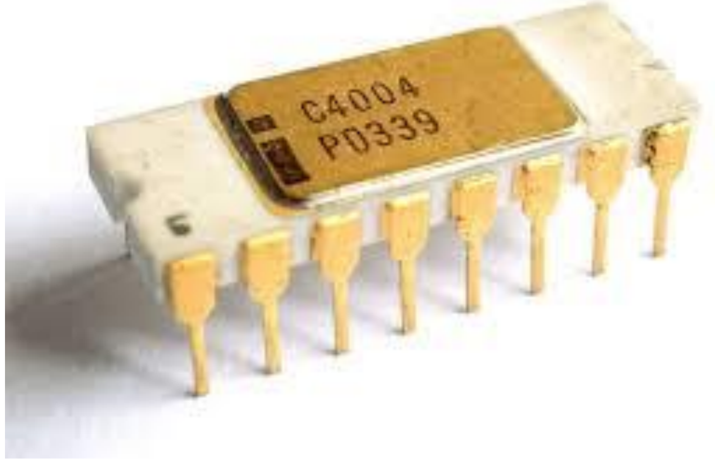
```
READY
10 FOR X=1 TO 10
20 PRINT "HOLA WIKIPEDIA"
30 NEXT X
RUN
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA

READY
█
```

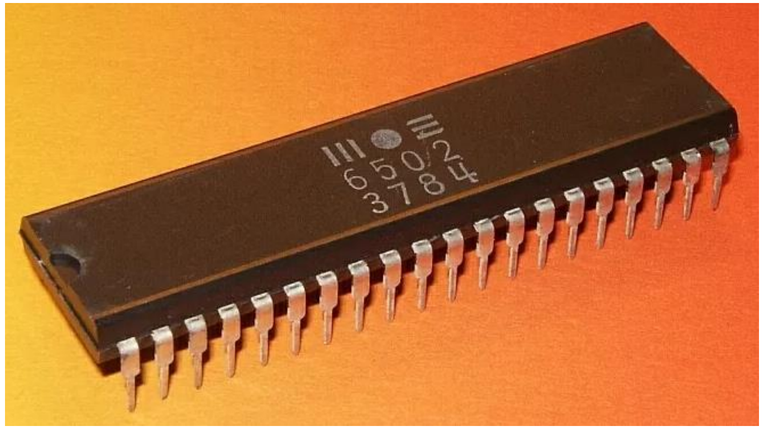
BASIC

→ Fourth Generation Computers

- 1971 - 1980
- Very Large Scale Integrated (VLSI) circuits (Microprocessors)
- Again reduce the size (called Microcomputers)
- Fast accessing & processing speed
- Eg : IBM PC and Apple II



First Microprocessors



IBM PC

Apple II

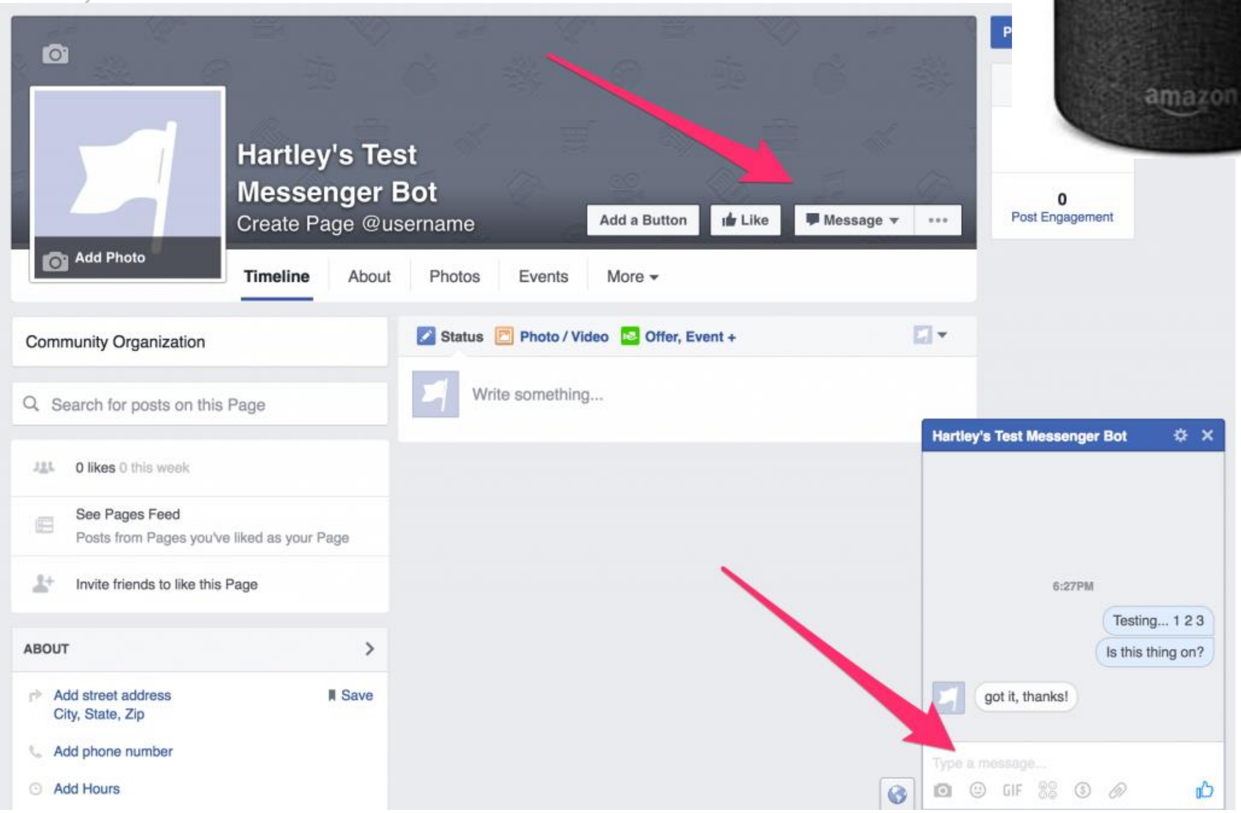
→Fifth Generation Computers

- 1982 -
- Ultra Large Scale Integration(ULSI) circuits

  (Microprocessors)
- Based on parallel processing hardware

  and AI (Artificial Intelligence) software
- AI Programming Language : LISP and Prolog
- Aim : Neural Language input and Self Learning capability
- Eg : Alexa , Robot, Google assistance , chat bot

Facebook Chatbot



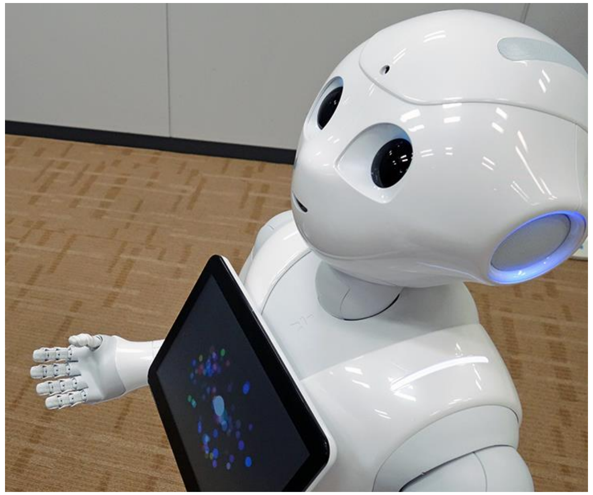
6:27PM

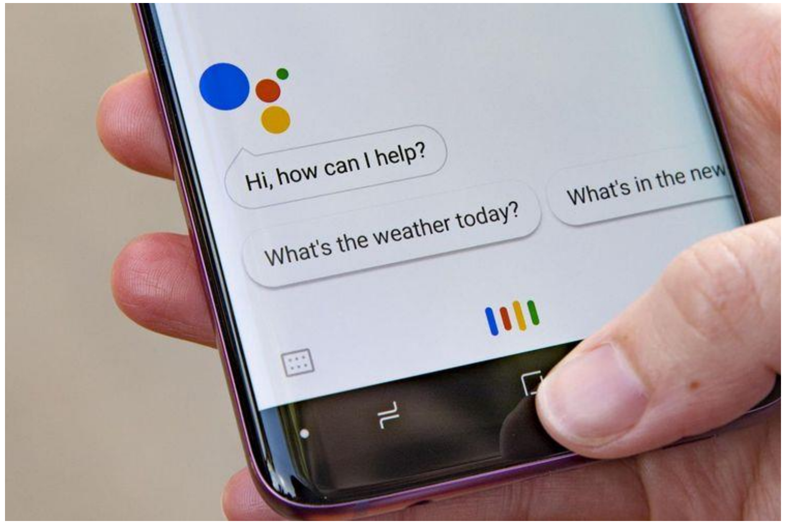
Testing... 1 2 3

Is this thing on?

got it, thanks!

ALEXA





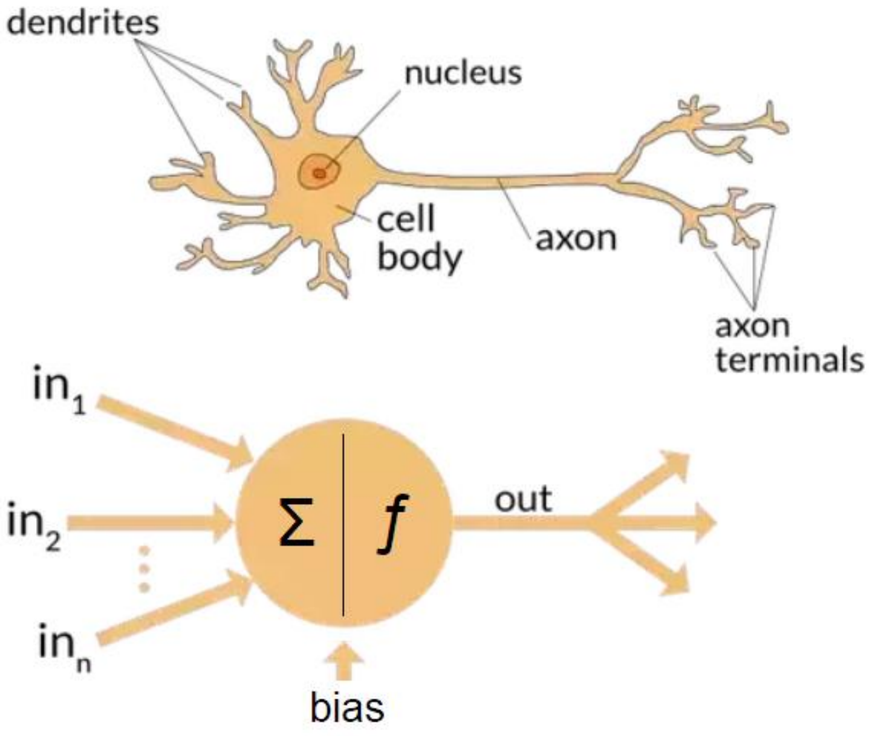
Hi, how can I help?

What's the weather today?

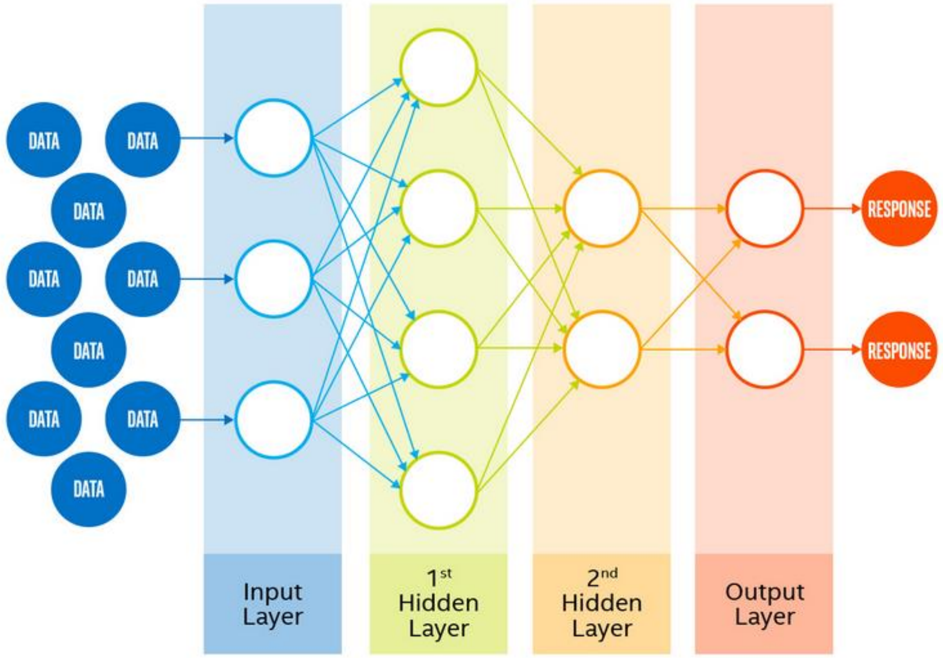
What's in the new

google assistant

AI



Neural Network



Evolution of Computing

- → Programming Language
  - → Machine Language
  - → Assembly Language
  - → High Level Language
  - → Algorithm

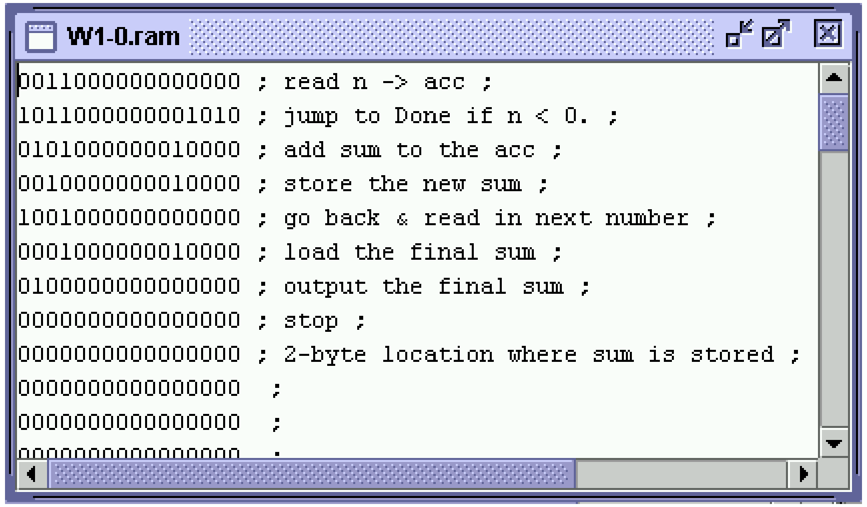
- → Theory of computing
  - → Alan Turing & Turing machine

→ Programming Language

- Is a formal language to communicate with machine
- Used to create program and control the behaviour of machine

→ Machine Language

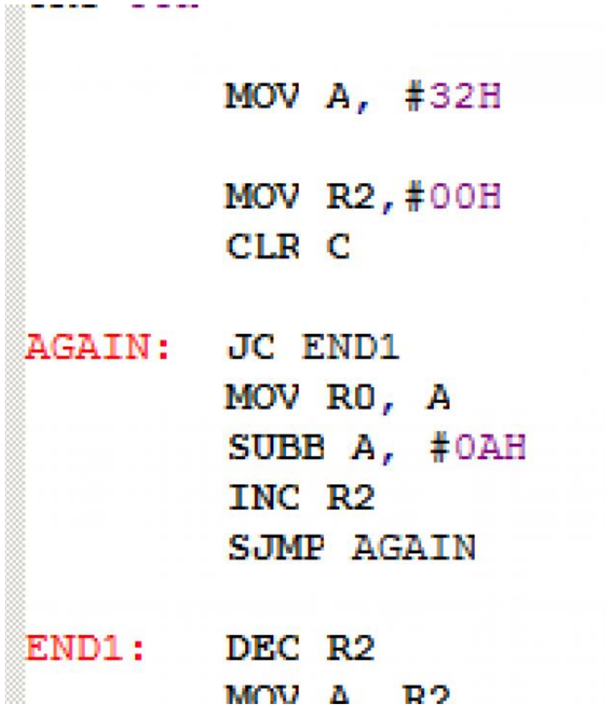
- First programming language developed
- String of binary digits(0 and 1)
- Difficult to program and error detection
- Programmer have good knowledge about computer working



```
0011000000000000 ; read n -> acc ;
1011000000001010 ; jump to Done if n < 0. ;
0101000000010000 ; add sum to the acc ;
0010000000010000 ; store the new sum ;
1001000000000000 ; go back & read in next number ;
0001000000010000 ; load the final sum ;
0100000000000000 ; output the final sum ;
0000000000000000 ; stop ;
0000000000000000 ; 2-byte location where sum is stored ;
0000000000000000  ;
0000000000000000  ;
0000000000000000  ;
```

→ Assembly Language

- English like words
- Assembler present
- easy to program and error detection better than machine language
- Machine dependent language (Not transferable from one computer to other)

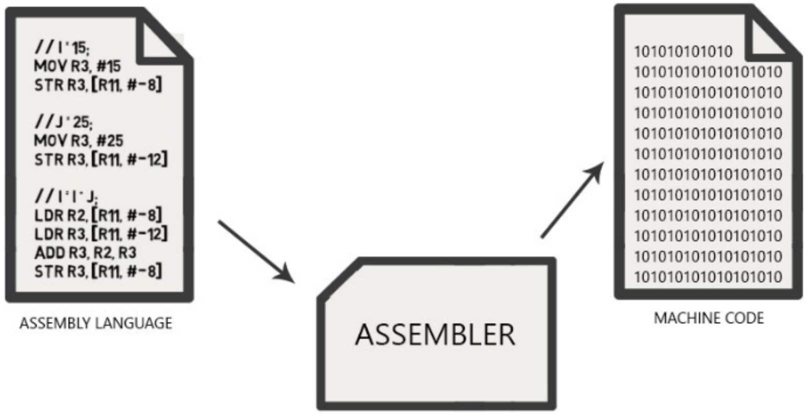


```
            MOV A, #32H

            MOV R2,#00H
            CLR C

AGAIN:      JC END1
            MOV R0, A
            SUBB A, #0AH
            INC R2
            SJMP AGAIN

END1:       DEC R2
            MOV A, R2
```



```
//I'15;
MOV R3, #15
STR R3, [R11, #-8]

//J'25;
MOV R3, #25
STR R3, [R11, #-12]

//I'I'J;
LDR R2, [R11, #-8]
LDR R3, [R11, #-12]
ADD R3, R2, R3
STR R3, [R11, #-8]
```

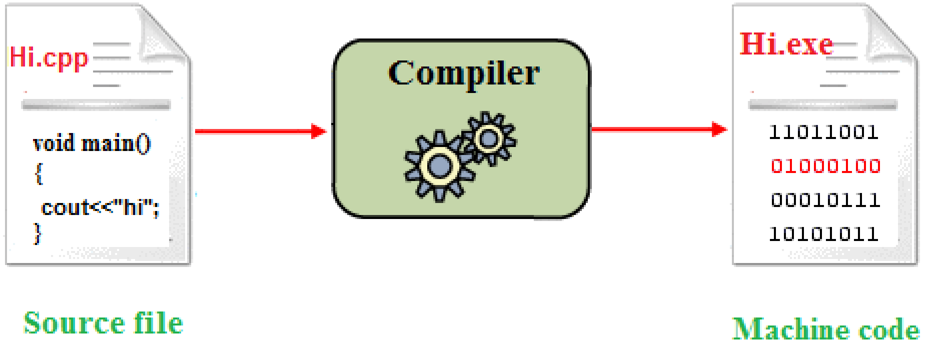
ASSEMBLY LANGUAGE

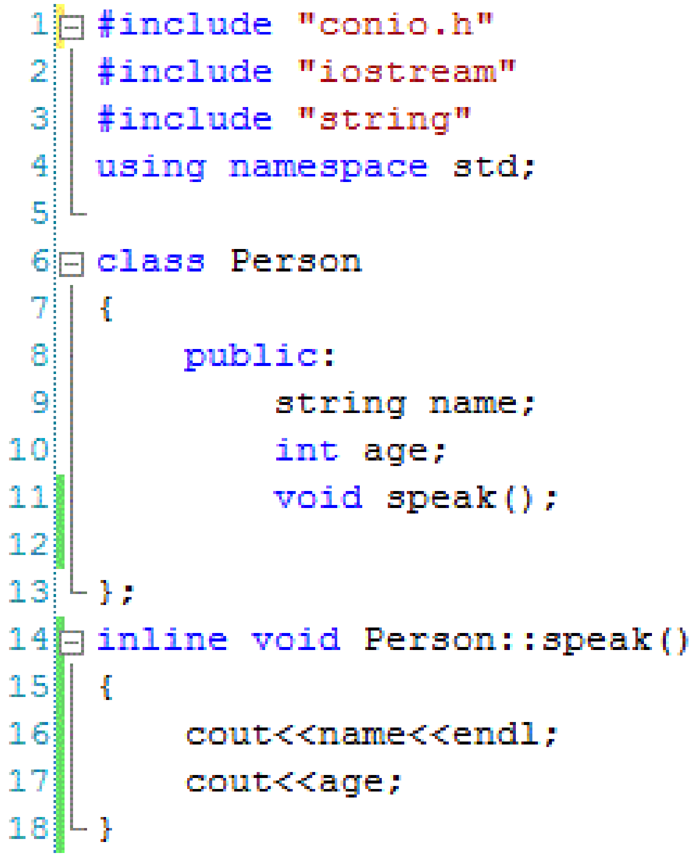
```
101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
101010101010101010
```

MACHINE CODE

→ High Level Language

- English like words and statements
- Compiler
- easy to program and error detection better
- Easy to learn program
- No need knowledge about computer working
- Machine Independent language





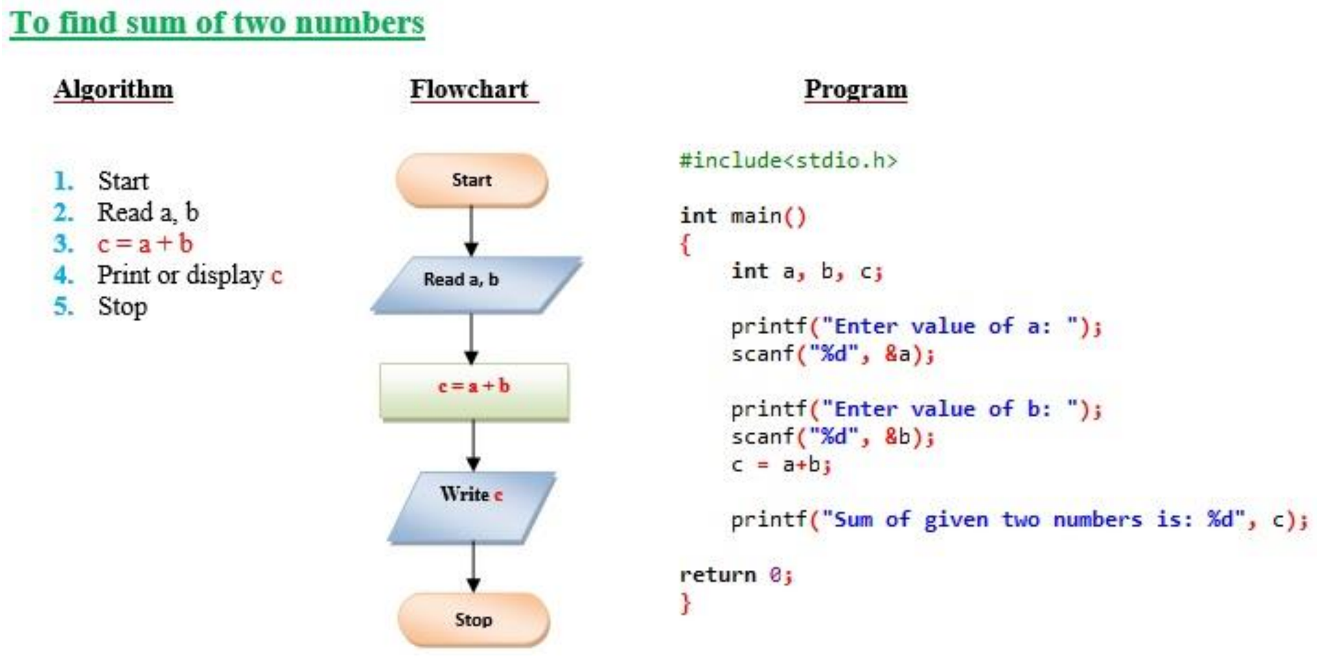
```
#include "conio.h"
#include "iostream"
#include "string"
using namespace std;

class Person
{
    public:
        string name;
        int age;
        void speak();

};
inline void Person::speak()
{
    cout<<name<<endl;
    cout<<age;
}
```

→ Algorithm

- Step by step solution of a problem
- First step of a program
- The steps can be converted in to different languages easily
- Language Independent



To find sum of two numbers

Algorithm

1. Start
2. Read a, b
3. c = a + b
4. Print or display c
5. Stop

Flowchart

Program

```
#include<stdio.h>

int main()
{
    int a, b, c;

    printf("Enter value of a: ");
    scanf("%d", &a);

    printf("Enter value of b: ");
    scanf("%d", &b);
    c = a+b;

    printf("Sum of given two numbers is: %d", c);

return 0;
}
```

→ Theory of computing

- Is the branch of programming that deals with :
- how effectively solve problems based on computation model and algorithm
- Also called computer science

→ Alan Turing & Tuning Machine

- He develop a machine called **Enigma Encrypting Machines**
- he came up with the idea : Universal state machine
- He design the 1st programmable computer
- Called **Father of Modern computer**

Teaching materials Prepared by

RIDHUN DEV

Lecture IHRD College

+1 / +2 Computer Science

8089552581