

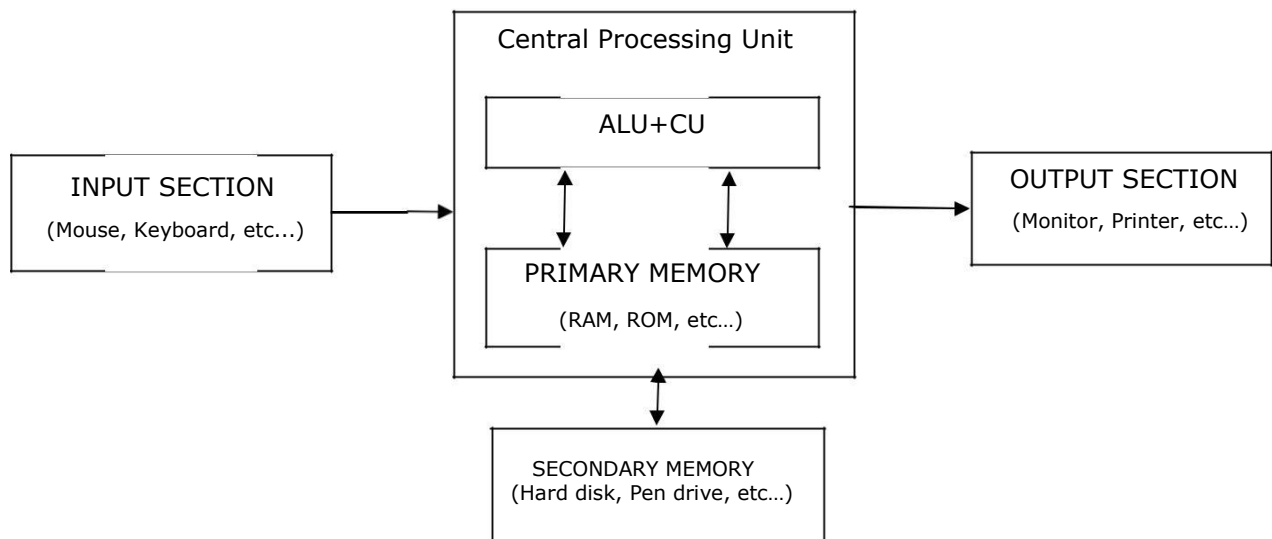
Computer Programming and Utilization (CPU) – 2110003

A) Computer Fundamentals

1 Draw the block diagram of computer architecture and explain each block.

Computer is made up of mainly four components,

- 1) Central processing unit (CPU)
- 2) Input section
- 3) Output section
- 4) Storage devices



1) Central Processing Unit (CPU):-

- Central processing unit is a main part of the computer system.
- It contains electronics circuitry that processes the data based on instructions.
- It also controls the flow of data in the system.
- It is also known as brain of the computer.
- CPU consists of,

Arithmetic Logical Unit (ALU)

- ✓ It performs all arithmetic calculations and takes logical decision.
- ✓ It can perform add, subtract, multiply, compare, count, shift and other logical activities.
- ✓ It calculates very fast.
- ✓ It takes data from memory unit and returns data to memory unit, generally primary memory.

Control Unit (CU)

- ✓ It controls all other units in the computer system.
- ✓ It manages all operations
- ✓ It reads instruction and data from memory.

Primary Memory:-

- ✓ It is also known as main memory.
- ✓ The processor or the CPU directly stores and retrieves information from it.
- ✓ This memory is accessed by CPU, in random fashion
- ✓ Generally currently executing programs and data are stored in primary memory
- ✓ Its storage capacity is very small compared to secondary storage.
- ✓ It is very fast in an operation compared to secondary storage
- ✓ RAM is Random Access Memory and it is volatile in nature.
- ✓ ROM is Read Only Memory and it can hold data permanently.
- ✓ PROM is Programmable Read Only Memory and it can hold data permanently. Programmer can store information only once. Modification is not allowed.
- ✓ EPROM is Erasable Programmable Read Only Memory. It can hold data permanently.
Programmer can delete and write on it again and again.

2) Input Section:-

- The devices used to enter data in to computer system are called input devices.
- It converts human understandable input to computer controllable data.
- CPU accepts information from user through input devices.
- Examples: Mouse, Keyboard, Touch screen, Joystick etc...

3) Output Section:-

- The devices used to send the information to the outside world from the computer is called output devices.
- It converts data stored in 1s and 0s in computer to human understandable information
- Examples: Monitor, Printer, Plotter, Speakers etc...

4) Storage devices (Secondary memory):-

- Secondary memory is also called Auxiliary memory or External memory.
- User can store data permanently.
- It can be modified easily.
- It can store large data compared to primary memory. Now days, it is available in

Terabytes.

- Examples: Hard disk, Floppy disk, CD, DVD, Pen drive, etc...

2 Describe advantages and limitations of computer. Or Explain characteristics of computer. Advantages

- **Speed:** It can calculate millions of expressions within a fraction of second. The micro second and nano second units are used to measure the speed of computers. There are some problems which cannot be solved within specified time limit without computer.
- **Storage:** It can store data in large quantity using various storage devices. Millions of paper file's data can be stored in single small pen drive. Moreover, it reproduces data whenever we need and whatever format we need. Now a days, Gigabytes and Terabytes are units of data storage devices.
- **Accuracy:** Computer performs the computations at very high speed without any mistakes. For example, multiplication of two very large number takes more time for human and there are high probabilities of mistakes. Computer does it in parts of the second with accuracy level we want. Very high level of accuracy is must in financial transaction, medical surgery, nuclear plant, etc... which can be satisfied by computer only.
- **Reliability:** It is very reliable device. The information stored in computer is available after years in same form. It works 24 hours without any problem as it never feels tiredness.
- **Automation:** Once the one task is created in a computer, it can be repeatedly performed again by single click whenever we want. For example, once the software for banking application is installed in a computer, it computes the interest by sending one command.

Limitations

- **Lack of intelligence:** It cannot think while doing work. It does not have natural intelligence. It cannot think about properness or effect of work it is doing. It can only execute the instructions but it cannot think about the correctness of these instructions.
- **Unable to Correct Mistakes:** It cannot correct the mistakes by itself. So if we have provided wrong or incorrect data then it produces wrong results or performs wrong calculations.

3 Describe various types of computer languages and mention its advantages and disadvantages.

Computer languages may be classified in three categories,

1) Machine level language or Low level language:-

- Computer directly understands this language. It is a language of 0's and 1's (binary).

Every CPU has its own machine language.

- **ADVANTAGES:**

1. It is very fast in execution
2. It does not require any extra system or software to run the program.
3. Translation is not required.
4. Suitable for low volume applications.

- **DISADVANTAGES:**

1. Programs are long and difficult to write and understand for human.
2. Debugging is very difficult task.
3. It is not portable.
4. Programmer requires detailed knowledge of architecture of microprocessor.

2) Assembly language:-

- Every machine language instruction is assigned to English word MNEMONIC such that it should describe function of instruction.
- System cannot understand this language directly so we require translator that convert assembly language to machine language. This translator is called assembler.
- Example : 8086 Instruction Set

- **ADVANTAGES:**

1. Programs are easy to understand compared to machine level language.
2. Programs are smaller in size compared to machine level language.
3. Programs can be entered quickly using alphanumeric keyboard.

- **DISADVANTAGES:**

1. It is not portable.
2. Programmer should know structure of assembly language of microprocessor.
3. It requires assembler as a translator.

3) Higher level language:-

- We can write programs in English like manner and it is more convenient to use.
- Programmer can perform complex task by using high level languages with less efforts.
- It is similar to natural language and mathematical notation.

ADVANTAGES:

- 1) Easier to learn.
- 2) Requires less time to write.
- 3) Provides better documentation.
- 4) Easier to maintain.
- 5) It is portable.

DISADVANTAGES:

- 1) It requires compiler or interpreter to convert higher level language to machine language.
- 2) Programmers need to learn structure of high level language.
- 3) It is bit slow compared to low level and medium level language.

4 Why C is called middle level language?

C is called middle level language because

- Syntax and keywords of C are just like higher level language (English).
- It gives advantages of higher level language through function, modular programming and breakup.
- It gives access to the low level memory through Pointers.
- Moreover it does support the Low Level programming i.e., Assembly Language.
- We can develop application specific programs in C and at the same time we can use features of assembly level language to give more speed and efficiency
- It is not hardware or system dependent. Hence portable programs can be written with C compiler.

5 Write a short note on types of software.

A set of instruction in a logical order to perform a meaningful task is called program and a set of program is called software.

System Software

System software is designed to operate the computer hardware efficiently. It provides and maintains a platform for running application software. Since system software runs at the most basic level of computer, it is called "low-level" software.

System software can be classified into three categories

- 1) Operating system: It controls hardware as well as interacts with users, and provides different services to user. It is a bridge between computer hardware and user.
Ex: Windows XP, Linux, UNIX, etc...
- 2) System support software: It makes working of hardware more efficiently. For example drivers of the I/O devices or routine for socket programming, etc...
- 3) System development software: It provides programming development environment to programmers.
Ex: Editor, pre-processor, compiler, interpreter, loader, etc.....

Application software

Application software is designed to help the user to perform general tasks (word processing, web browser ...) or some specific task (accounting, ticket booking ...).

Example: Enterprise software, Accounting software, Office suites, Graphics software and

media players.

Application software is classified into two categories.

- **General purpose software:**

It is used widely by many people for some common task, like word processing, web browser, excel, etc... It is designed on vast concept so many people can use it.

- **Special purpose software:**

It is used by limited people for some specific task like accounting software, tax calculation software, ticket booking software, banking software etc... It is designed as per user's special requirement.

6 Define the following terms:

1. Program

A set of instruction in a logical order to perform a meaningful task is called program.

2. Software

A set of instruction in a logical order to perform a meaningful task is called program, and set of program is called software.

3. Hardware

A physical part of computer is known as Hardware. User can see and touch the hardware components.

4. Operating system

Operating system is system software which works as an interface between hardware and user and provides interactive platform to user.

5. Assembler

Assembler is system software which converts lower level programs of assembly language to machine language.

6. Compiler

It translates programs of higher level language to machine language. It converts whole program at a time.

7. Interpreter

It translates programs of higher level language to low level programs. It converts program line by line.

8. Linker

It bind symbolic code of source and library file to make executable Program.

7 Give Difference between Interpreter and Compiler

Interpreter	Compiler
It scans the program line by line and stops scanning whenever error occurs.	It scan the whole program at a time and display number of errors together if any.
It converts the source program	It converts the whole source code into object code at a time
for every execution to run, source code is required	After compilation source code is not required.

Execution is slower than compiler	Execution is faster than Interpreter
the object code is generated when the program is error free.	Interpreter generates object code for each line immediately if it is error free.

9. Give Difference between System Software and Application Software.

System Software	Application Software
System software comprises of those programs, which directs the computer	Application software is the software developed for solving business problem
System software varies from computer to computer	Application software varies from organization to organization
System software written in low level language	Application software written in high level or object oriented language
Detail knowledge of hardware is required	This requires detail knowledge of organization
System software is used to improve performance and maximum utilization of system resources	Application software is used to improve the speed and quality of business activity
The manufacturers along with the hardware usually supply system software	Application software is developed by individual or programmer or supplied by software vendors as per requirement.

Flowchart & Algorithm

1 What is Algorithm? What is Flowchart? Write down the advantages and disadvantages. Compare them.

Flowchart

Flowchart is a pictorial or graphical representation of a program. Each step in the process is represented by a different symbol and contains a short description of the process step. The flow chart symbols are linked together with arrows showing the process flow direction. This pictorial representation can give step-by-step solution of the given problem.

Advantages

- Easy to draw.
- Easy to understand logic.
- Easy to identify mistakes by non computer person.
- Easy to show branching and looping.

Disadvantages

- Time consuming.
- Difficult to modify.
- Very difficult to draw flowchart for big or complex problems.

Algorithm

An Algorithm is a finite sequence of well defined steps by steps procedure for solving a problem in systematic manner which produces effective output for specific input in given amount of time. It is written in the natural languages like English.

Advantages

- Easy to write.
- Human readable techniques to understand logic.
- Algorithms for big problems can be written with moderate efforts.

Disadvantages

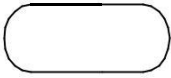
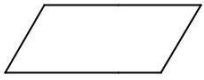

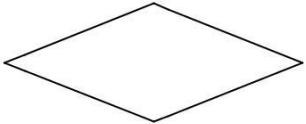
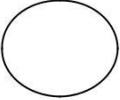
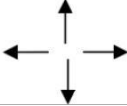
- Difficult to debug.
- Difficult to show branching and looping.
- Jumping (goto) makes it hard to trace some problems.

Comparison of algorithm and flow chart

Flowchart	Algorithm
It is a pictorial representation of a process.	It is step wise analysis of the work to be done.
Solution is shown in graphical format.	Solution is shown in non computer language like English.

Easy to understand as compared to algorithm.	It is somewhat difficult to understand.
Easy to show branching and looping.	Difficult to show branching and looping.
Flowchart for big problem is impractical	Algorithm can be written for any problem

Symbols used in flowchart:

	Start / Stop
	Input / Output (Read / Print)
	Process
	Decision making
	Connector
	Arrows

Write an algorithm and Draw a flowchart

- 1. To find whether given number is even or odd. Algorithm:-

Step 1: Start

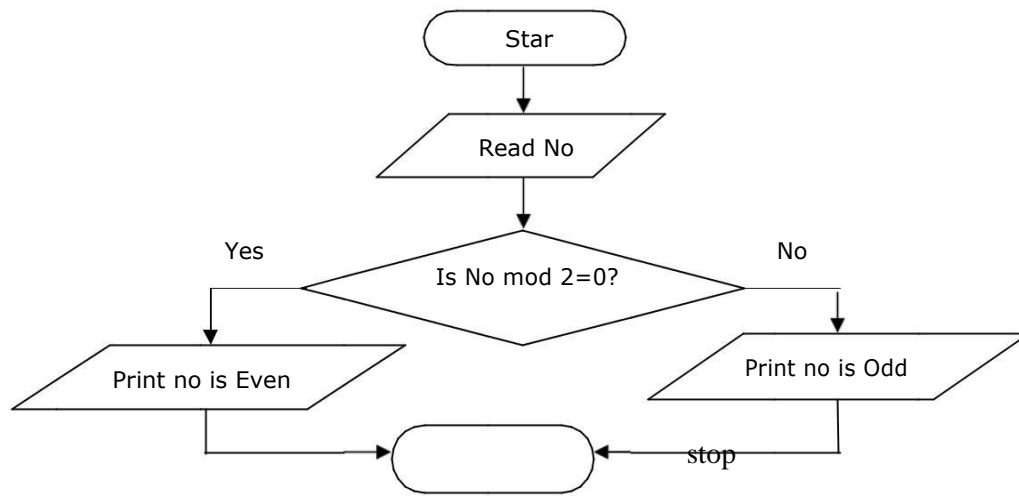
Step 2: Input no

Step 3: If no mod 2=0, then goto next step else go to step no. 5

Step 4: Print given no is Even go to last step

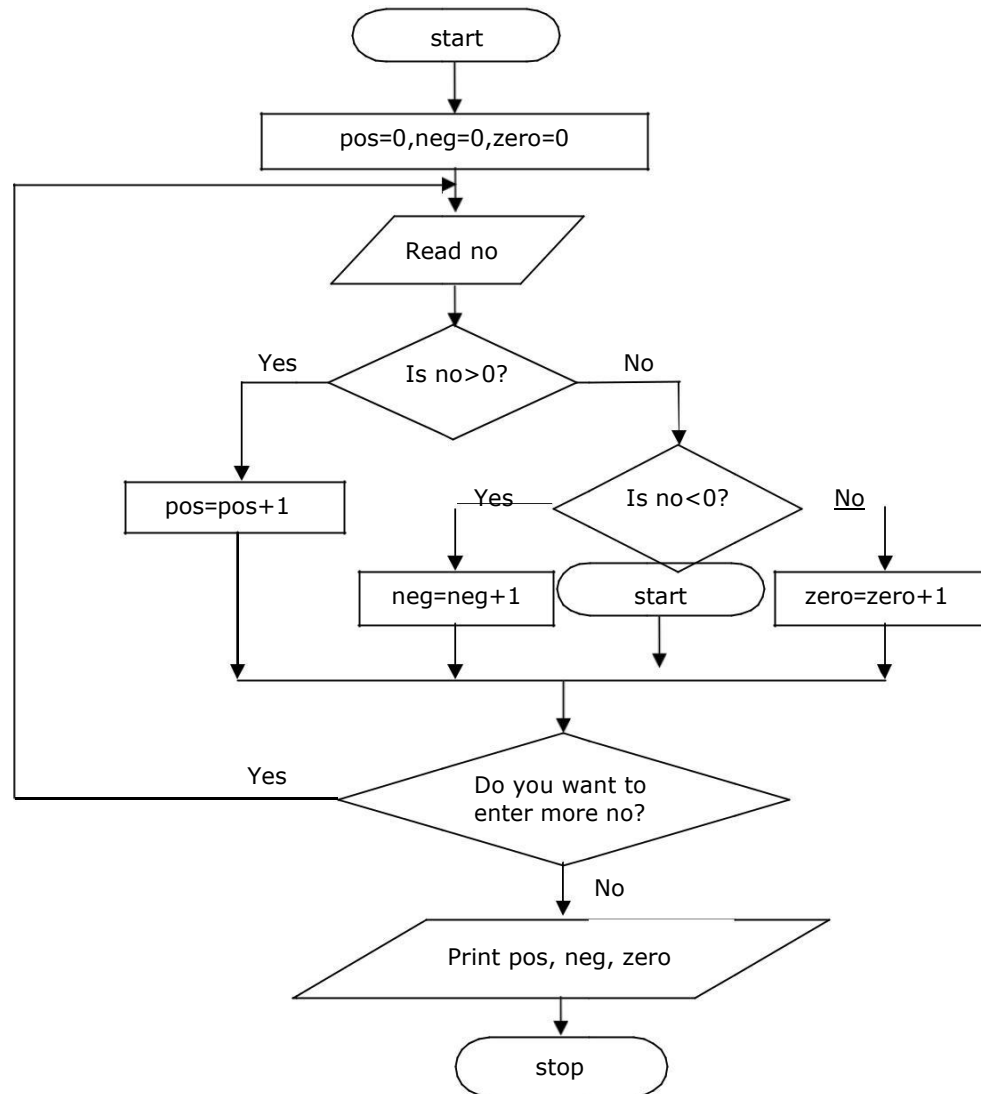
Step 5: Print given no is odd

Step 6: Stop



- To enter number still user wants and at the end it should display count of total number of positive, negative and zero entered.

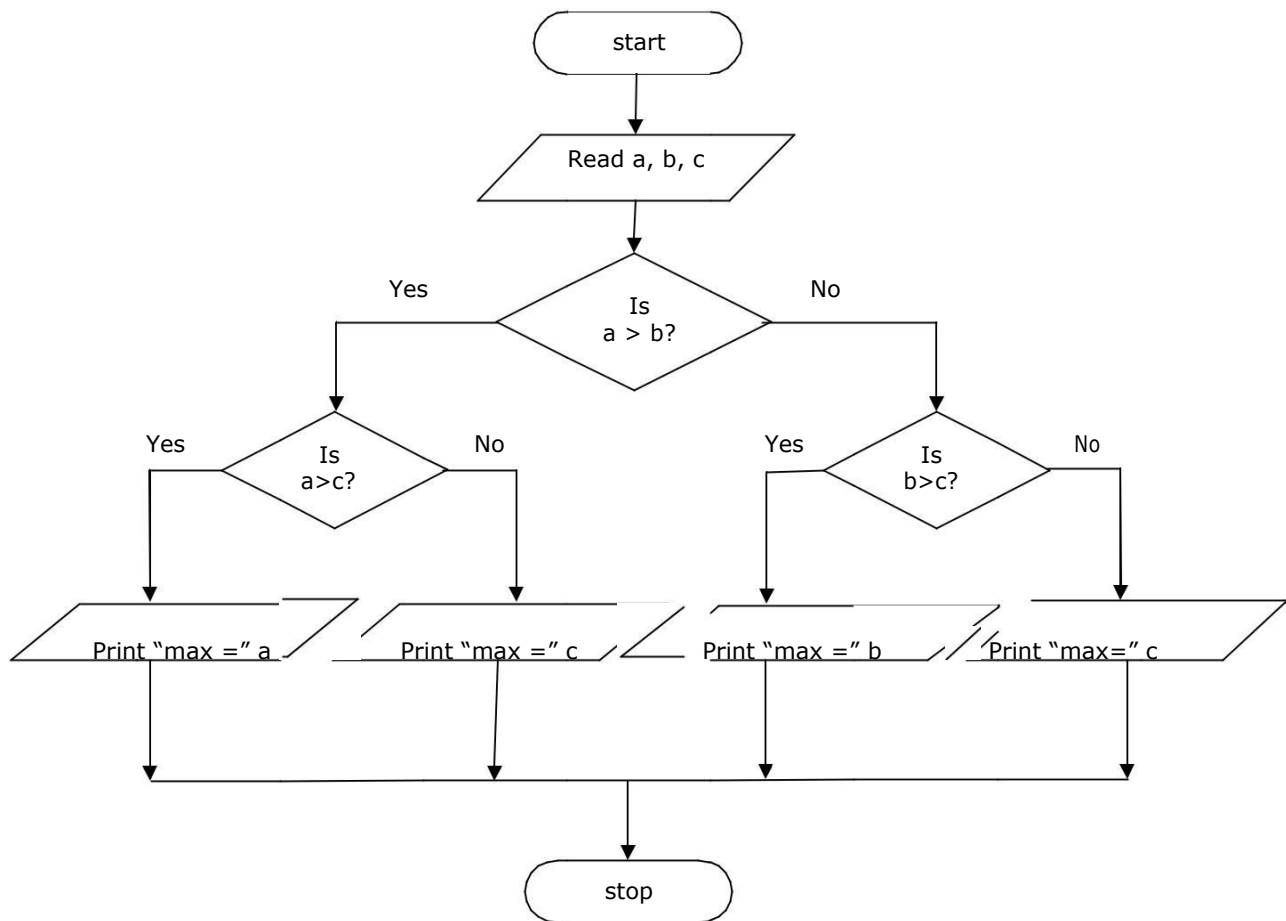
Step 1 : Start
Step 2 : Initialize pos=0,neg=0,zero=0
Step 3: Read no
Step 4 : If no>0,then goto 7
Step 5 : If no<0,then goto 8
Step 6 : Increment zero by 1, zero=zero+1.go to 9
Step 7 : Increment pos by 1, pos=pos+1.go to 9
Step 8 : Increment neg by 1, neg=neg+1.go to 9
Step 9 : Print “Do you want to enter more number?”
Step 10 : Read ans
Step 11 : If ans = “y”, then goto 3
Step 12 : Print pos, neg, zero.
Step 13: Stop.



- To print maximum number from a given 3 numbers.

Step 1: Start
 Step 2: Read a,b,c
 Step 3: if a > b then go to next step else go to step no 7
 Step 4: if a > c then go to next step else go to step no 6
 Step 5: print a is maximum and go to step no 10
 Step 6: print c is maximum and go to step no 10
 Step 7: if b > c then go to next step else go to step no 9
 Step 8: print b is maximum and go to step no 10
 Step 9: print c is maximum and go to step no 10
 Step 10 : Stop

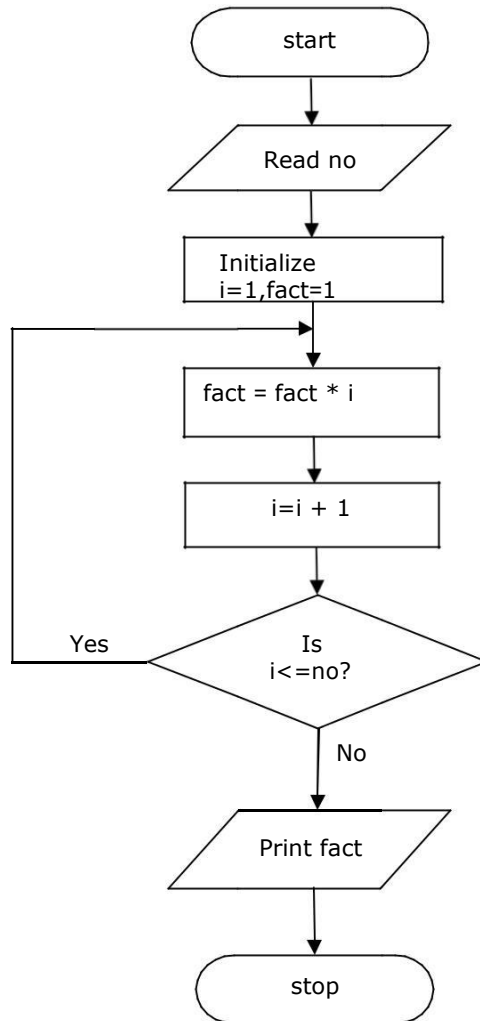
Flowchart:-



- To find the factorial of a given number.

Step 1: Start
Step 2: initialize $i=1$, $fact=1$
Step 3: Read no
Step 4: Repeat step no 4 to step no 6 till $i \leq no$
Step 5: Calculate $fact=fact*i$
Step 6: calculate $i=i+1$
Step 7: print "factorial=" fact
Step 8: Stop

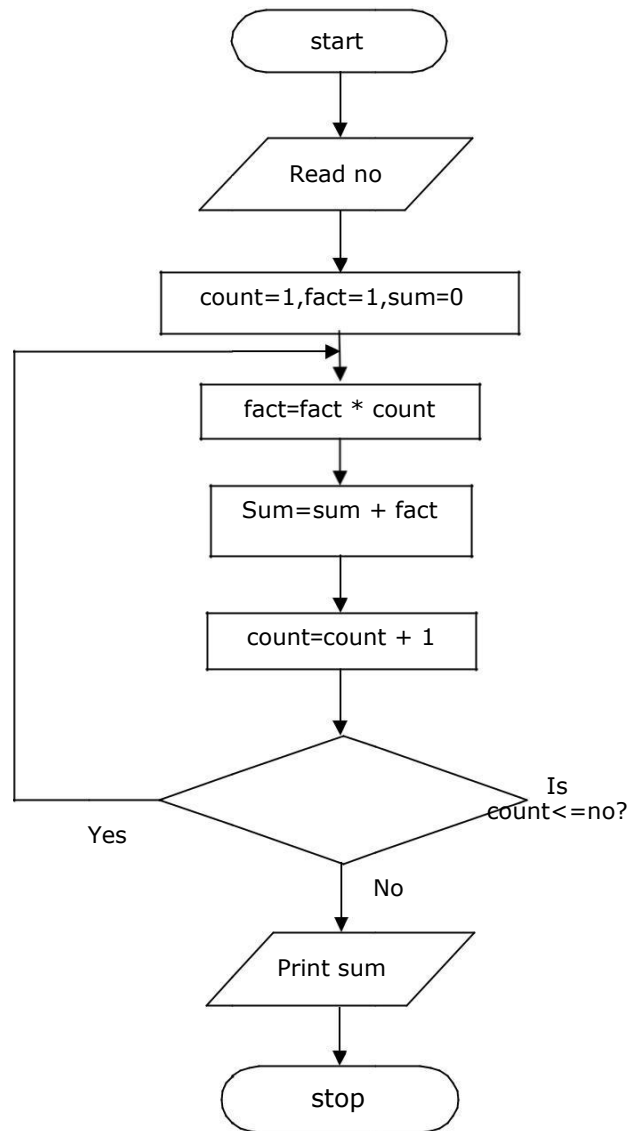
Flow chart



- To solve series $1!+2!+3!+4!+\dots+n!$

Algorithm:

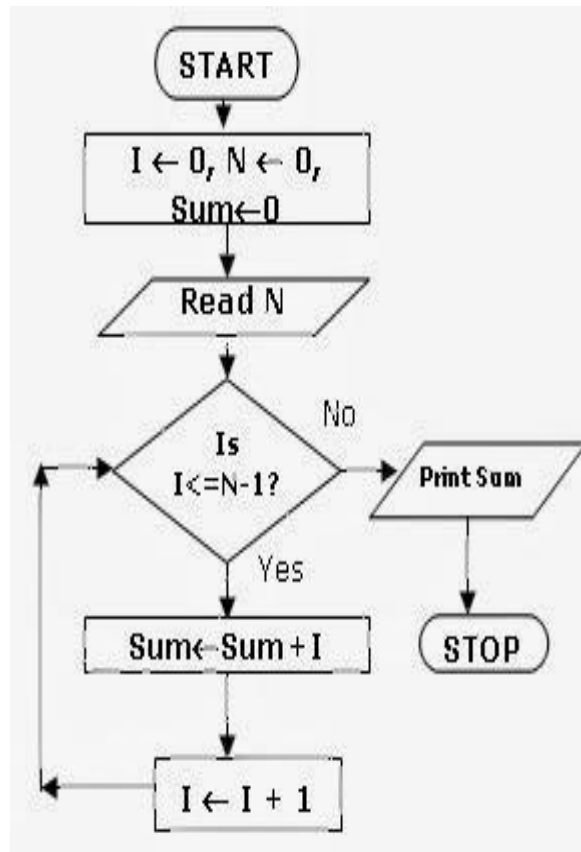
Step 1 : Start
 Step 2 : Initialize count=1, fact=1, sum=0
 Step 3 : Read no
 Step 4: Calculate fact=fact*count
 Step 5: Calculate sum=sum + fact.
 Step 6: Increment count by 1, count=count+1.
 Step 7: Repeat step no 3 to 6 till count<=no
 Step 8: print sum
 Step 9: Stop



- To print sum of n numbers.

Algorithm:

Step 1: start
 Step 2 : initialize i=0, sum=0
 Step 3: Read n
 Step 4: repeat step 4 to step no 6 till i < n
 Step 5: sum = sum + i
 Step 6: increment value of i, i=i+1
 Step 7: print "Sum =" sum
 Step 8: stop

Flow chart

- To reverse a given number

Algorithm

Step 1: Start

Step 2: [Initialize] $Rev = 0$

Step 3: Read: number

Step 4: repeat step no 4 to step no 7 till number > 0

Step 5: Compute $r = \text{number} \% 10$

Step 6: Computer $Rev = Rev * 10 + r$

Step 7: Decrement value of number, $\text{number} = \text{number} / 10$

Step 5: Print " Rev Num =" Rev

Step 6: Stop

Flow chart

