27.3.2020

MODULE : 5 BASIC ALGORITHMS

Algorithm: is a finite list of instructions used to perform a task.

The computer is used to solve the real-life problems which are first identified and then designed before being programed for a computer. The Algorithms are used to designed the solution of a problem. Therefore, we can say that an algorithm is a tool for solving any computational problem.

1

ALGORITHM:

An algorithm is a finite sequence of instructions, a logic and explicit step-by-step procedure for solving a problem starting from a known beginning. An algorithm produce the same output information given the same input information, and several short algorithms can be combined to perform complex tasks such as writing a computer program.

- without algorithm we cannot solve or design flowchart.
- with out algorithm we cannot write any program. "Algo" is very important.

For example :-

Q. Algo to create e-mail Address.

step 1 : Start

step 2 : Create a variable to receive the user's email address.

step 3 : clear the variable in case it's not empty.

step 4 : store the response in the variable

step 5 : check the stored response to see if it is a valid email address.

step 6 : End

Example

Problem : To find the roots of the quadratic eqn.

Input : Given quadratic equation ($ax^2+bx+c=0$).

Output : Roots of the quadratic equation (x_1 and x_2).

Sequence of steps :

1. READ a, b and c
2. Calculate $D = b^2-4ac$
3. CHECK if ($D<0$)
   a. PRINT "root are imaginary"
   b. EXIT

   otherwise
   a. CALCULATE roots x1 and x2 using the formula

   $x1 = (-b+\sqrt{(b^2-4ac)})/(2a)$

   $x2 = (-b-\sqrt{(b^2-4ac)})/(2a)$

   b. PRINT roots x1 and x2
4. EXIT

Q2 Write an Algorithm to add two numbers.

Step 1: start;
step 2: Input the first number x;
step 3: Input the second number y;
step 4: set z = x + y;
step 5: print z;
step 6: end;

. This algorithm performs the steps in a purely sequential order.

Q3 Write an algorithm to set the equality of two nos

by: Suvesh Kr. Shukla

Step 1: start;
step 2: Input the 1st no. A;
step 3: Input the 2nd no. B;
step 4: if A = B
then print "Equal"
else
print "Not equal"
step 5: End;

for exercises

Q Write an Algorithm to calculate Simple Interest.

Q Write an Algorithm to calculate sum of first ten even nos.

Problem : linear Search (array, size, key)

Input : An sorted Array, size of the array and the search key

output : location of the key (if found, otherwise wrong location

1. Begin
2. for i := 0 to size -1 do
   if array[i] = key then
   return i
3. Done
4. Return invalid location
5. End.

Example

input

0	1	2	3	4	5	6	7
20	4	89	75	10	23	45	69

Key = 10

output : Item found at location 4

Binary search: When the list is sorted we can use the binary search technique to find items on the list. In this procedure, the entire list is divided into two sublist. If the item is found in the middle position it returns the location, otherwise jumps to either left or right sublist and do the same procedure again until finding the item or exceed the range.

ALGORITHM

binary search (array, start, end, key)

Input: An sorted array, start and end locations, and the search key.

output: location of key (if found) otherwise wrong location

```
Begin
  if start <= end then
    mid := start + ( end - start ) / 2
      if array [mid] = key then
          return mid location
      if array [mid] > key then
```

~~call binary search (array, start, mid, key)~~

P.T.O.

cont.

call Binary search(array, mid+1, end, key)

else when array[mid] < key then

call binary search(array, start, mid-1, key)

else

return invalid location

End.

Example

Input: Sorted list of data:

0	1	2	3	4	5	6	7
12	25	48	52	67	79	88	93

Key = 79

Output : found at location : 5

Sorting refers to arranging data in a particular format. Most common orders are in numerical or lexicographical order.

The importance of sorting lies in the fact that data searching can be optimized to a very high level, if data is stored in a sorted manner.

Bubble sort : It is a comparison based sorting algorithm. In this algorithm adjecent elements are compared and swapped to make the correct sequence.

Algorithm

bubbleSort (array, size)

Input : An array of data, and the total no. in the array

Output : The sorted Array

```
Begin
  for i:= 0 to size-1 do
      flag := 0;
      for j := 0 to size-i-1 do
        if array[j] > array[j+1] then
          swap array[j] with array[j+1]
          flag := 1
  done
```

Cont..... P.T.O.

```
    if flag ≠ 1 then
        break the loop.
    done
End.
```

for Example

Input: A list of unsorted data:

56 98 78 12 30 51

Output: After sorting:

12 30 51 56 78 98

Insertion Sort: This sorting technique is similar with the card sorting technique, in other words, we sort card using using insertion sort mechanism. For this technique, we pick up one element from the data set and shift the data elements to make a place to insert back the piked up an element into the data set.

Algorithm

insertionSort (array , size)

Input: An array of data, and the total number in the array

output: The sorted array

```
Begin
  for i := 1 to size-1 do
      key := array[i]
      j := i
      while j > 0 AND array[j-1] > ked do
          array[j] := array[j-1];
          j := j-1
      done
      array[j] := key
  done
End
```

Input: The unsorted list:

9 45 23 71 80 55

Output: Array after sorting:

9 23 45 55 71 80

Selection Sort: In this technique list is divided into two parts. In one part all elements are sorted and in another part the items are unsorted. At first, we take the maximum and minimum data from the array. After geting the data we place it at the begining of the list by replacing the data of first place with the minimum data. After performing the array is getting smaller.

Algorithm

Selection_Sort (array , size)

Input : An array of data, and the total number in the array

Output : The sorted Array

```
Begin
  for i: = 0 to size - 2 do
        i Min : = i ;
      for j : = i+1 to size - 1 do
       if array [j] < array [i Min] then
          i Min : = j
      done
      swap array [i] with array [i Min].
  done
End
```

for Example

Input : The unsorted list :

5 9 7 23 78 8

output : Array before sorting :

5 9 7 23 78 8

Array After sorting :

5 7 8 9 23 78

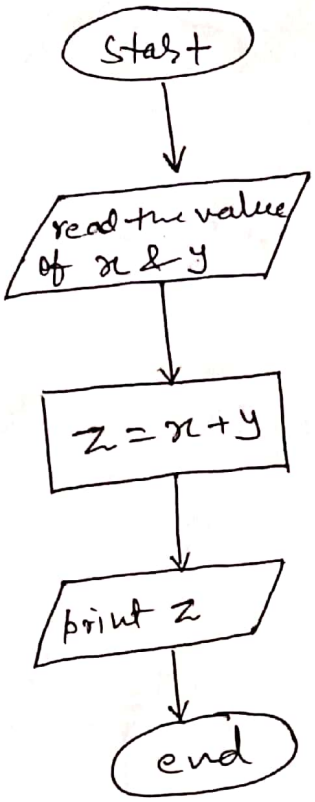
FLOW CHART:

It is a pictorial representation of an algorithm. As flow chart is a picture of work to be done, it may be printed in form of symbol.

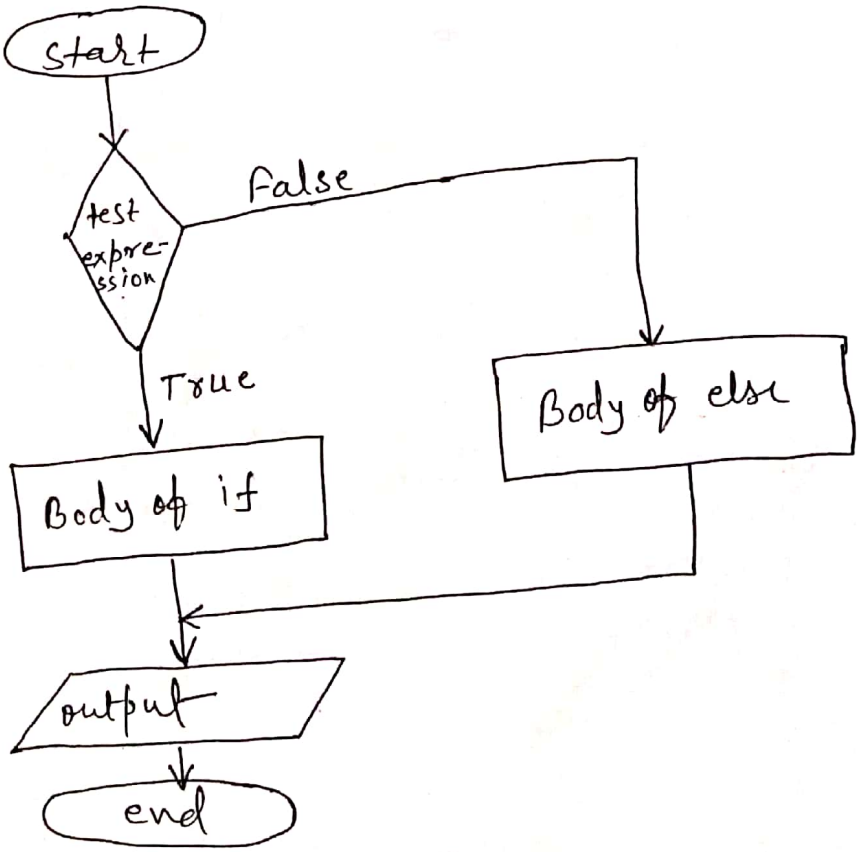
Symbols of flow chart:

Symbols	Name	Work
oval	Start/End/terminal	to start or end the flowchart
Arrow →	flow	to flow control of flowchart
Parallelogram	input/output	used for input or output
diamond	decision	used for YES/NO or TRUE/FALSE decision
rectangle	processing	used for activity as addition, division etc.
circle	connector	to connect two or more complex flow diagrams.
	Direct Data	used for database
	Document	to use documentation

Q. Write or draw a flowchart to add two numbers.

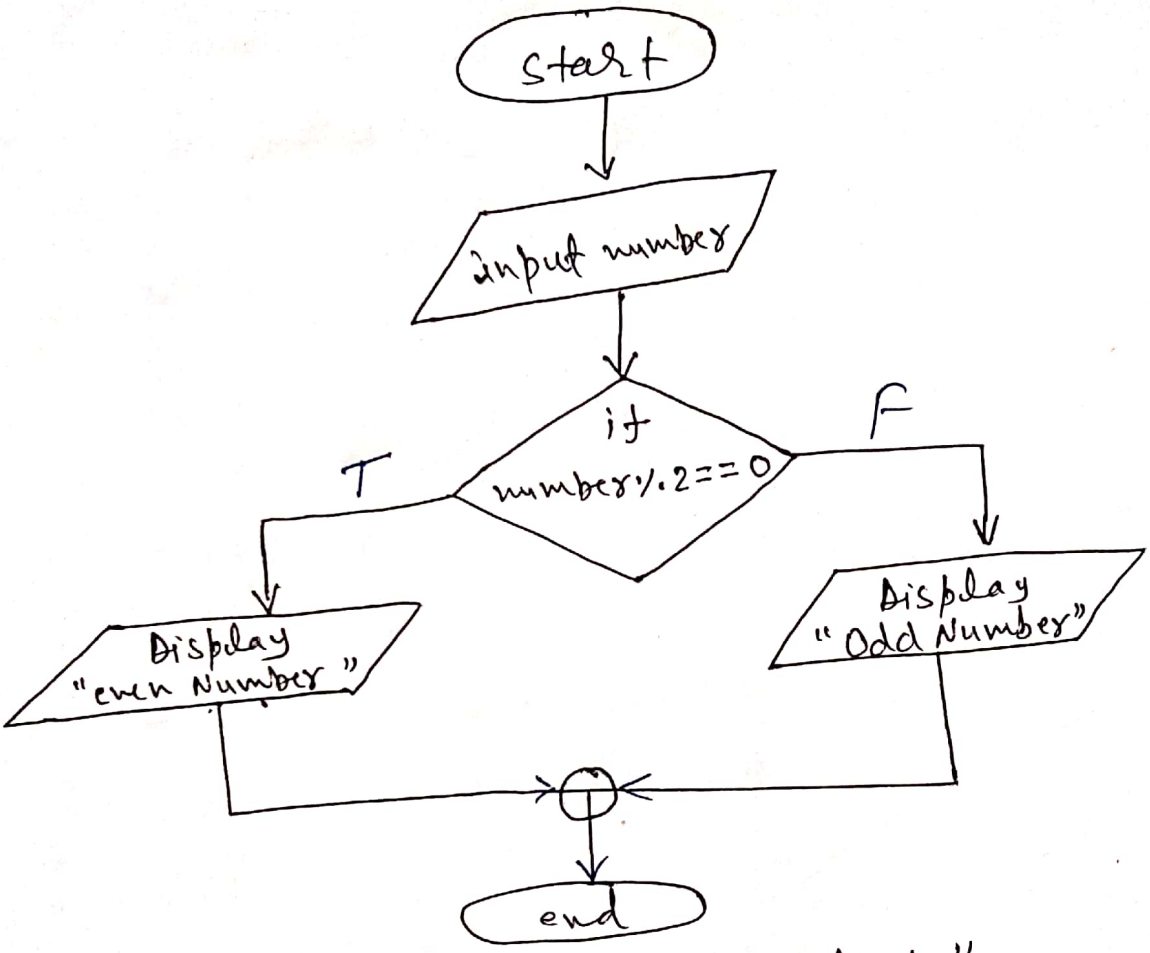


Q. Draw a flow chart for <u>if-else</u>.



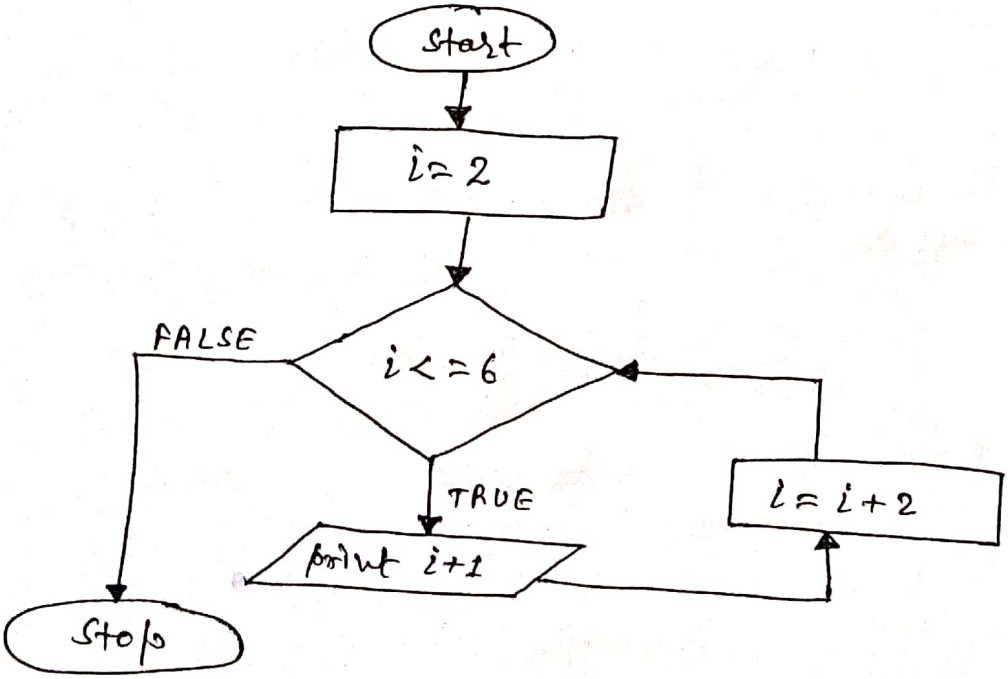
Q. Draw Flowchart to check odd even number.

P-13



by:
S.K.Shukla

Q. Draw a flowchart for "for loop".



PSEUDOCODE

Pseudocode:- It is an artificial and informal language that helps programmers develop algorithm. It is a text-base detail of algo. or it is a form of structured english that describe algorithm. P-14

pseudocode

Q. Pseudo code for making a cup of tea.

by:- S.K. Shukla

- Organise everything together;
- put in kettle;
- put teabag in cup;
- put water into kettle;
- wait for kettle to boil;
- add water to cup;
- Remove teabag with spoon/fork;
- Add milk and/or suger;
- serve;

Q. Consider the problem of seaching for an entry in a phone book with only condition.

- Get first entry;
- If it is the required entry (;) x
- then write down phone number;
- else get next entry (;) x
- If this is the correct entry (;) x
- then write down entry;
- else get next entry (;) x
- if this the correct entry (;) x
- .....;

Prof. Suresh Kumar Shukla