SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF INFORMATION TECHNOLOGY

PROBLEM SOLVING AND C PROGRAMMING

I YEAR - I SEM

UNIT 1 – Introduction to Problem Solving Techniques

TOPIC 6 – Simple Strategies for Developing Algorithms

SIMPLE STRATEGIES FOR DEVELOPING ALGORITHM

- They are two commonly used strategies used in developing algorithm
    1. Iteration
    2. Recursion

- **ITERATION:**

- The iteration is when a loop **repeatedly executes** till the controlling condition becomes false.
- The iteration is applied to the set of instructions which we want to get repeatedly executed.
- Iteration includes "initialization, condition, and execution" of statement within loop and update (increments and decrements) the control variable.

- A sequence of statements is executed until a specified condition is true is called iterations.
    1. for loop
    2. While loop

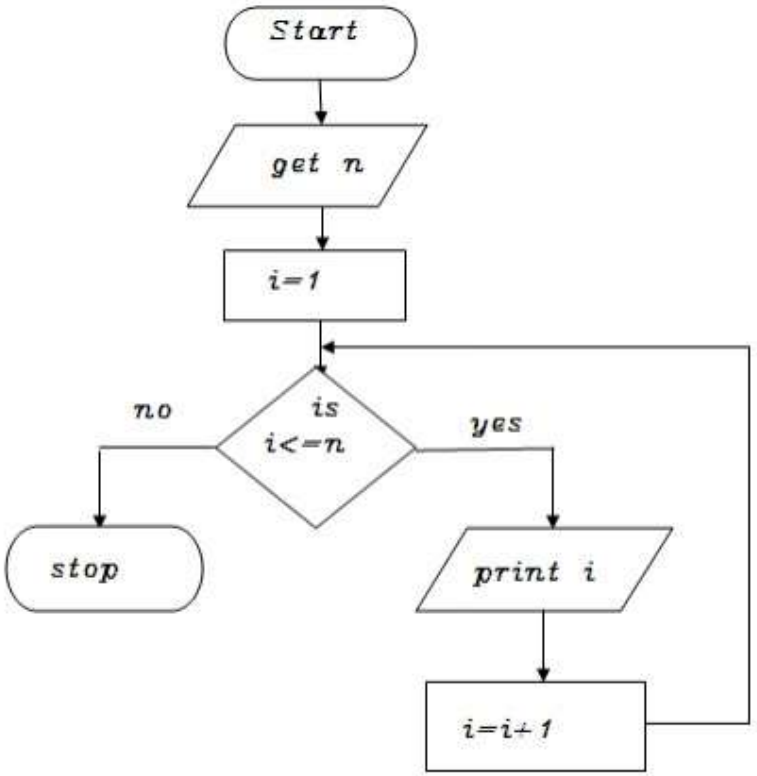
FOR & WHILE LOOP

Syntax for For:	Example: Print n natural numbers

FOR(*start-value* to *end-value)* DO
statement
... ENDFOR
	BEGIN
	GET n
	INITIALIZE i=1
	FOR (i<=n)DO
	PRINT i
	i=i+
	1
	ENDFOR
	END

Syntax for While:	**Example: Print n natural numbers**

WHILE (condition) DO
statement
...
ENDWHILE
	BEGIN
	GET n
	INITIALIZE i=1
	WHILE(i<=n) DO
	PRINT i
	i=i+1
	ENDWHILE
	END



RECURSION

- Recursions:
  - A function that calls itself is known as recursion.
  - Recursion is a process by which a function calls itself repeatedly until some specified condition has been satisfied.

- Algorithm for factorial of n numbers using recursion
- Main function:

  Step1: Start
  Step2: Get n
  Step3: call factorial(n)
  Step4: print fact
  Step5: Stop
- Sub function factorial(n):

  Step1: if(n==1) then fact=1 return fact
  Step2: else fact=n*factorial(n-1) and return fact

RECURSION

➢Pseudo code for factorial using recursion:

Main function:

```
BEGIN
GET n
CALL
factorial(n)
PRINT fact
END
```

Sub function factorial(n):

```
IF(n==1) THEN
        fact=1
        RETURN fact
ELSE
RETURN fact = n * factorial (n - 1)
```