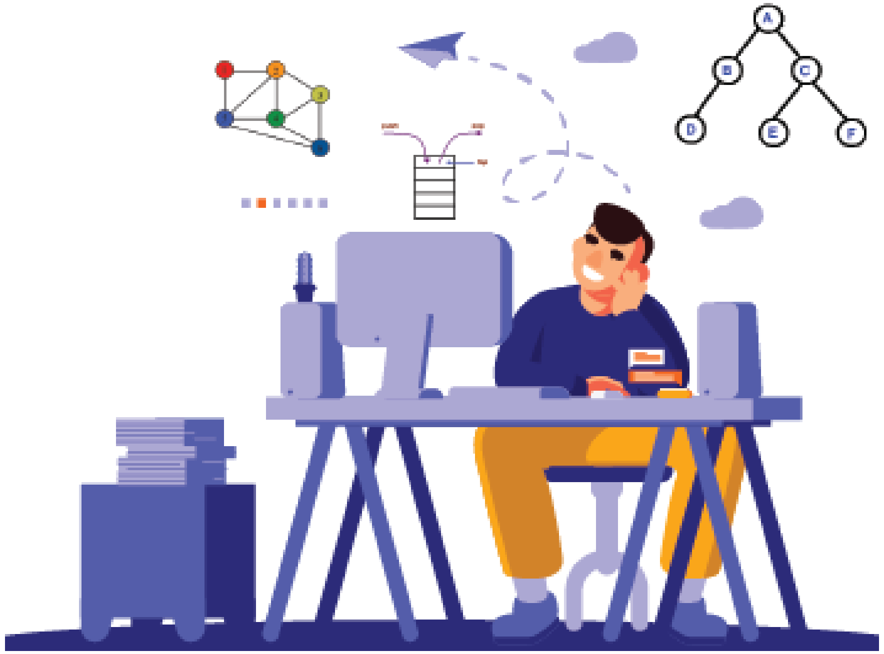
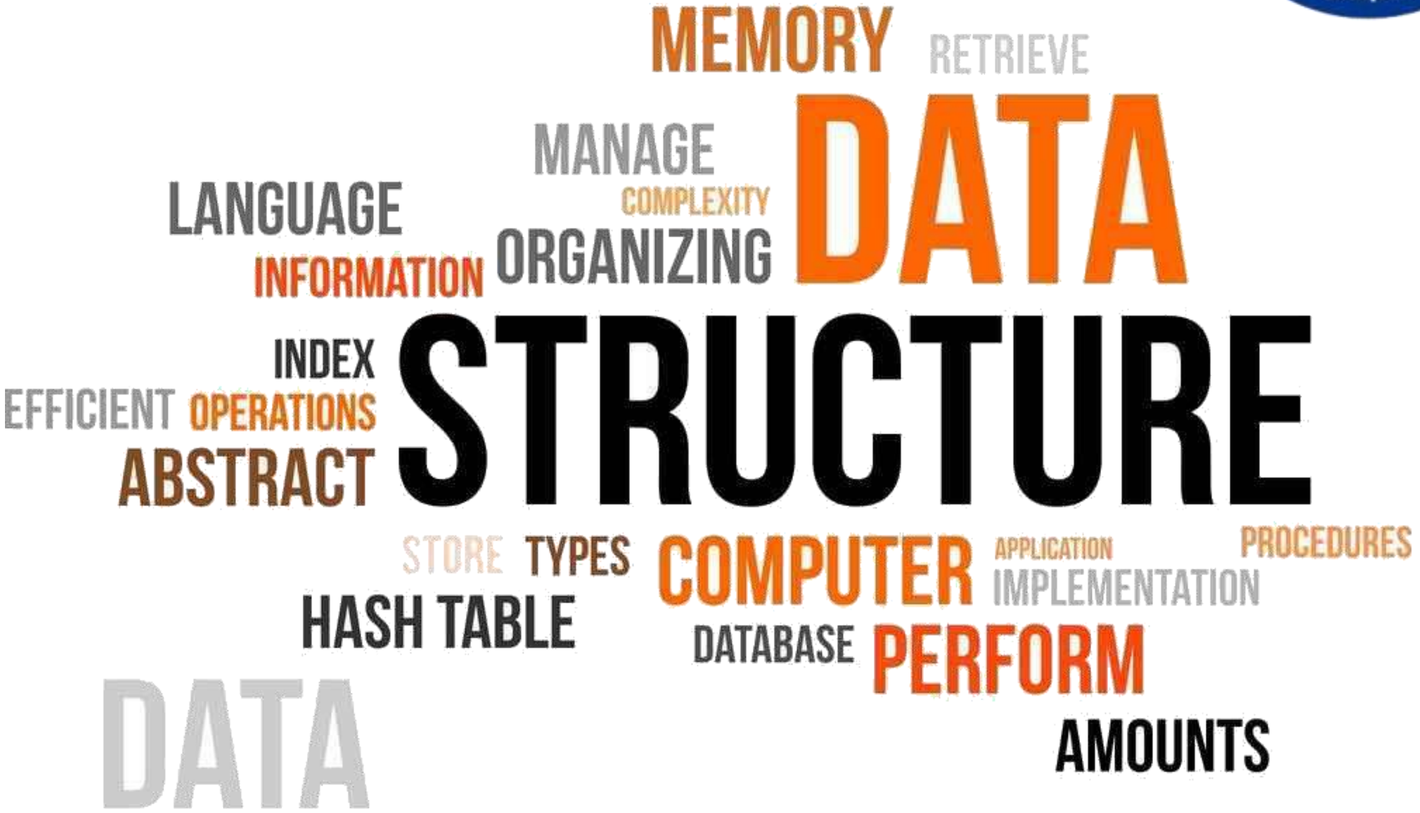


Data Structures

Course code: IT623



Dr. Rahul Mishra
Assistant Professor
DA-IICT, Gandhinagar

Lectures 3

Standard Notations and common functions

3. Modular function & Arithmetic

* For any integer a and any positive integer n, the value $a \bmod n$ is the remainder (or residue) of the quotient a/n.

read as: "a modulo n"

* More exactly $K \pmod M$ is the unique integer r such that

$$\boxed{K = Mq + r \quad \text{where} \quad 0 \le r < M.}$$

* When K is positive, simple divide K by M to obtain remainder r. Thus,

$25 \pmod 7 = 4$; $25 \pmod 5 = 0$, $35 \pmod{11} = 2$, $3 \pmod 8 = 3$

* If $(a \bmod n) = (b \bmod n)$, we write $\boxed{a \equiv b \pmod n}$ and say that a is equivalent to b, modulo n.

*

$\equiv >$ $\boxed{\text{Congruent}}$ $\rightarrow$

The mathematical congruence relation is defined as follows:

$a \equiv b \pmod M$ if and only if M divides $b-a$.

Standard Notations and common functions

* In other words, $a \equiv b \pmod{n}$ if a and b have the same remainder when divided by n.

* Equivalently, $a \equiv b \pmod{n}$, if and only if n is a divisor of $b-a$.

* $a \not\equiv b \pmod{n}$ if a is not equivalent to b, modulo n.

$\boxed{\text{4. Integer and Absolute value Functions}}$

* Let x be any real number. The integer value of x, written $INT(x)$, converts x into an integer by deleting (truncating) the fractional part of the number.

$INT(3.14) = 3$, $INT(\sqrt{5}) = 2$, $INT(-8.5) = -8$, $INT(7) = 7$

$INT(x) = \lfloor x \rfloor$ when x is positive

$INT(x) = \lceil x \rceil$ when x is negative.

* Similarly, absolute value gives a positive integer.

Standard Notations and common functions

5. Exponentials:

For all real $a > 0$, m, and n, we have following identities:

$$a^0 = 1,$$
$$a^1 = a,$$
$$a^{-1} = 1/a$$
$$(a^m)^n = a^{mn}$$
$$(a^m)^n = (a^n)^m$$
$$a^m a^n = a^{m+n}$$

For all n and $a \geq 1$, the function a^n is monotonically increasing in n.

Standard Notations and common functions

$$n! = \begin{cases} 1 & \text{if } n=0 \\ n \cdot (n-1)! & \text{if } n>0 \end{cases}$$

Thus, $n! = 1 \cdot 2 \cdot 3 \cdots n$

6. Factorial Function

$$n! = 1 \cdot 2 \cdot 3 \cdots (n-2)(n-1)n.$$

$$f^{(i)}(n) = \begin{cases} n & \text{if } i=0 \\ f(f^{(i-1)}(n)) & \text{if } i>0 \end{cases}$$

Normal recurrence relation, not factorial

For example, if $f(n) = 2n$, then $2^i n$.

7. Fibonacci number

$F_0 = 0,$

$F_1 = 1,$

$F_i = F_{i-1} + F_{i-2}$ for $i \geq 2$

0, 1, 1, 2, 3, 5, 8, 13, 21 ....

Standard Notations and common functions

$\boxed{\text{8. Polynomials}}$

Given a nonnegative integer d, a polynomial in n of degree d, is a function $p(n)$ of the form

$$p(n) = \sum_{i=0}^{d} a_i n^i \quad ; \quad \boxed{\text{sometimes } n \text{ is } x.}$$

where the constants $a_0, a_1, \dots a_d$ are the coefficients of the polynomial and $\boxed{a_d \neq 0.}$

* A polynomial is asymptotically positive if and only if $a_d > 0$

* For an asymptotically positive polynomial $p(n)$ of degree d, we have $p(n) = \Theta(n^d)$

* For any real constant $a \geq 0$, the function n^a is monotonically increasing

* For any real constant $a \leq 0$, the function n^a is monotonically decreasing.

* A function $f(n)$ is polynomially bounded if $f(n) = O(n^k)$ for some constant k.

Standard Notations and common functions

9. Logarithms

$\lg n = \log_2 n$ (binary algorithm)

$\ln n = \log_e n$ (natural logarithm)

$\lg^k n = (\lg n)^k$ (exponentiation)

$\lg \lg n = \lg(\lg n)$ (composition)

An important notational convention we shall adopt is that logarithm functions will apply one to the next term in the formula, so that $\lg n + k$ will mean $(\lg n) + k$ and not $\lg(n+k)$ ✗. If we hold $b > 1$ constant, then for $n > 0$, the function $\log_b n$ is strictly increasing.

Standard Notations and common functions

For all real $a>0$, $b>0$, $c>0$, and n

* $a = b^{\log_b a}$,

* $\log_c (a,b) = \log_c a + \log_c b$,

* $\log_b a^n = n \log_b a$,

* $\log_b a = \dfrac{\log_c a}{\log_c b}$,

* $\log_b (1/a) = -\log_b a$,

* $\log_b a = \dfrac{1}{\log_a b}$

* $a^{\log_b c} = c^{\log_b a}$

Standard Notations and common functions

* An algorithm, intuitively speaking, is a finite step-by-step list of well-defined instructions for solving a particular problem.

* Simplifed one : " An Algorithm is a sequence of instructions for solving a problem"

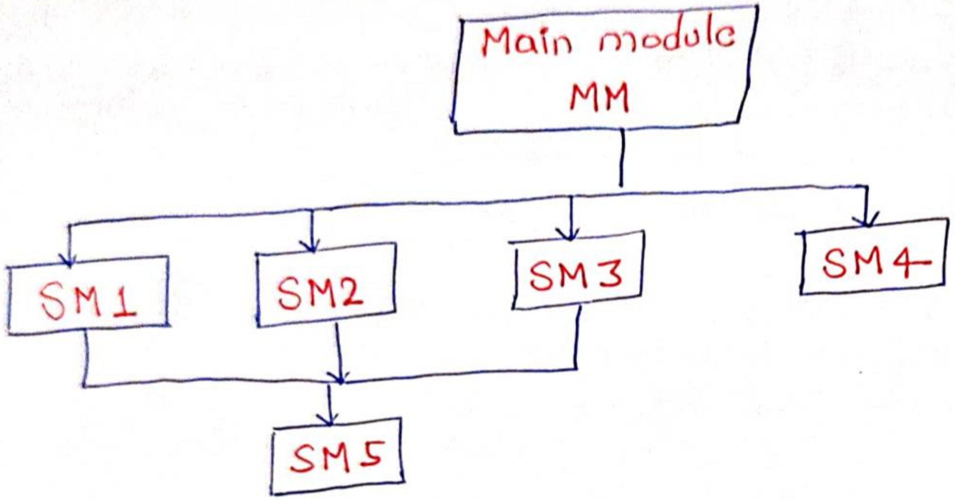
* Algorithms are implemented using programs.

* An efficient program is organized into modules
  - *> Main Module
  - *> Sub Module

* Main module: General description of the algorithm. (MM)

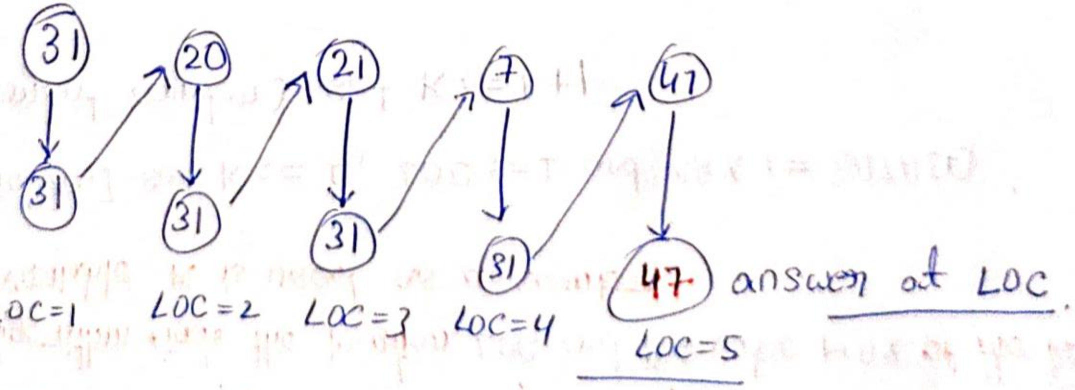
* Sub module: Detailed and specific information. (SM)

Standard Notations and common functions



Example – An array DATA of numerical values is in memory. We want to find the location LOC and the value MAX of the largest element of DATA.

eg:- [31, 20, 21, 7, 47]



Standard Notations and common functions

Algorithm : (Largest Element in Array) A non empty array DATA with N numerical values is given. This algorithm finds the location LOC and the value MAX of the largest element of DATA. The variable K is used as a counter.

Step 1. [Initialize.] Set K := 1, LOC := 1 and MAX := DATA[1]

Step 2. [Increment counter] Set K := K+1

Step 3. [Test counter.] IF K > N, then:

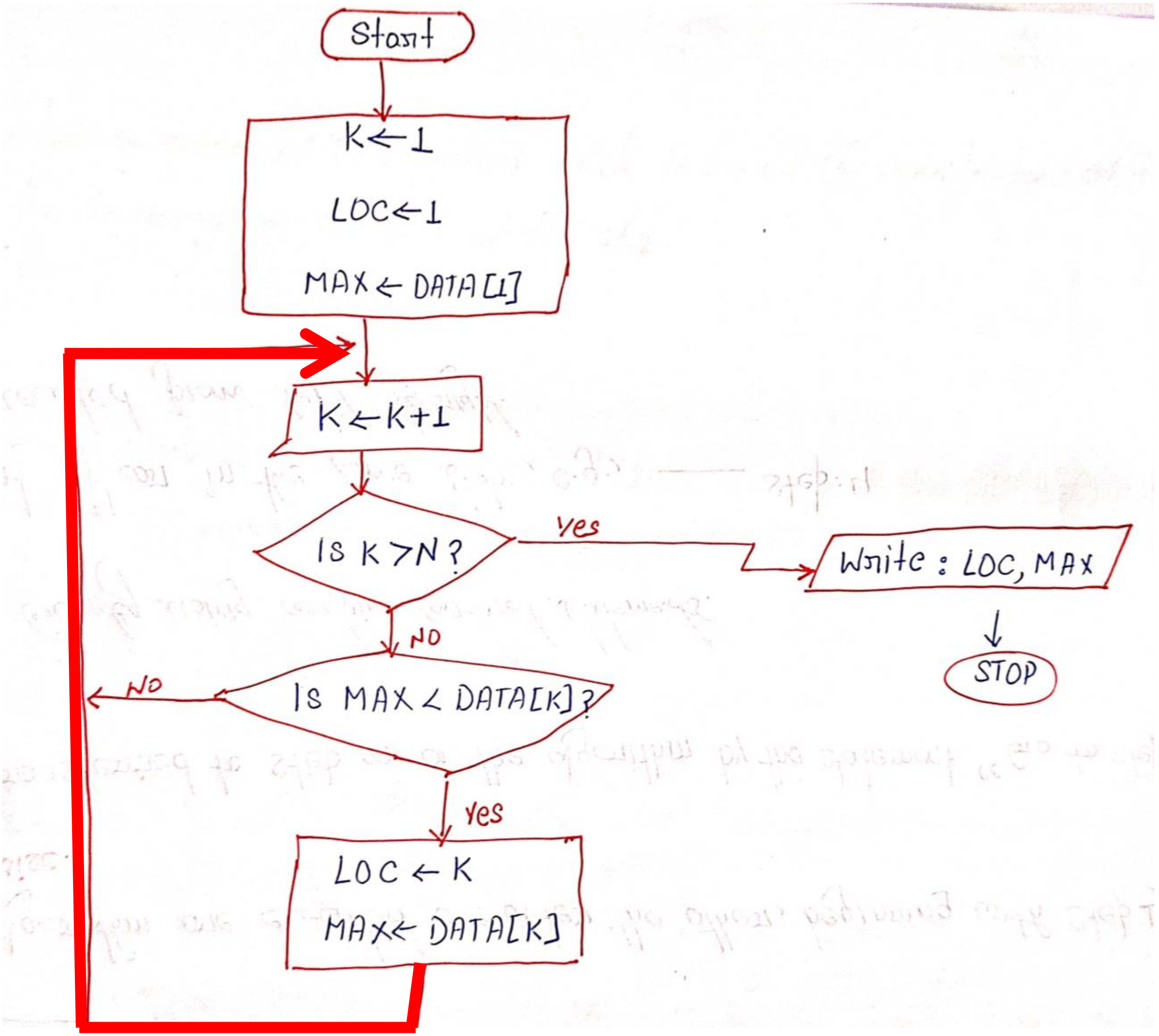
Write : LOC, MAX, and Exit

Step 4. [Compare and update.] IF MAX < DATA[K], then :

Set LOC : = K and MAX := DATA[K].

Step 5. [Repeat Loop.] Go to Step 2.

Standard Notations and common functions



Standard Notations and common functions

Steps, Control, Exit

* The steps of the algorithm are executed one after the other, beginning with step 1, unless indicated otherwise.
* Control may be transferred to step n of the algorithm by the statement "Go to step n" e.g. Step 5.
* We can eliminate Go by using certain control statements.
* If several statement appear in the same step, e.g., —— step: 4 They are executed from left to right.

* Exit completion

Standard Notations and common functions

Comments

Each step may contain a comment in brackets which indicates the main purpose of the step.
Usually appear at the begining or the end of the step.

Variable Names

* Variables names will use capital letters, as MAX and DATA
* Single - letter names of variables used as counters or subscript will also captalized in algorithm.
* Lower case can be used

Assignment statement

:= Pascal

←

=

all are used as per language.

Standard Notations and common functions

Input and output

Input: Read Variables names (scanf)

Output: Write Messages and/or variable names
(printf)

Procedures

→ Used for an independent algorithmic module which solves a particular problem.

$\underbrace{\text{Procedures / Modules}}_{\text{specific}}$ / $\overbrace{\text{Algorithms}}^{\text{Broad.}}$ → Interchangable

Standard Notations and common functions

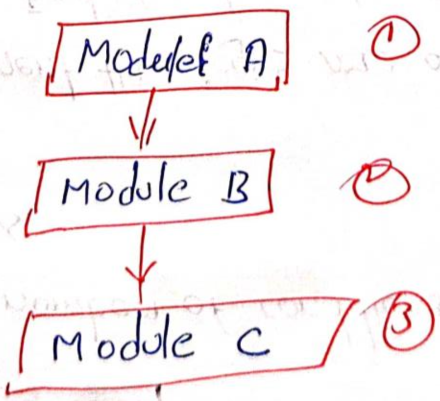
Control structures

Three types of logic, or flow of control, called

1) Sequence logic, or Sequential flow
2) Selection logic, or conditional flow
3) Iteration logic, or repetitive flow

→ Sequence logic discussed in previous algorithmic example

→ Unless instructions are given to the contrary, the modules are executed in the obvious sequence.



Standard Notations and common functions

Selection Logic (Conditional Flow)

* selection logic employs a number of conditions which lead to a selection of one out of several alternative modules.

* The structures which implement this logic are called conditional structure or IF statement.

e.g.: End of such a structure by the statement →

[End of IF structure]

1) Single Alternatives

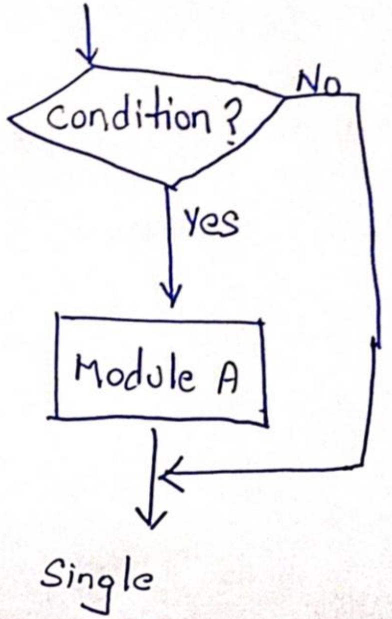
IF Condition, then:

[Module A]

[End of IF Structure]

2) Double Alternatives

3) Multiple Alternatives



Standard Notations and common functions

Double Alternatives

This structure has the form

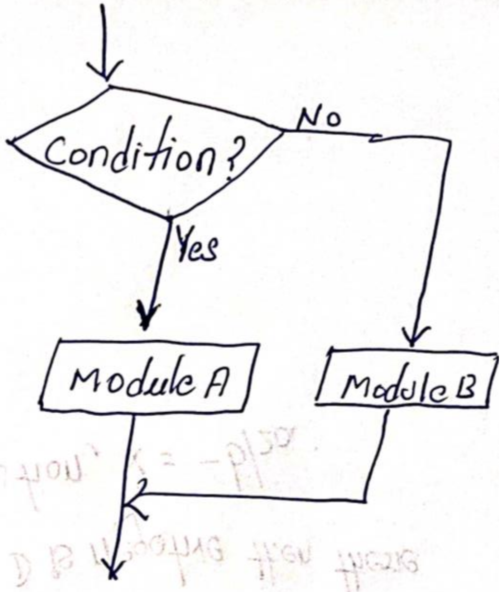
IF condition; then:

[Module A]

Else:

[Module B]

[End of IF structure]



Multiple Alternatives

This structure has the form

IF condition(1), then:

[Module A_1]

Else if condition(2), then:

[Module A_2]

⋮

Else if condition(M), then:

[Module A_M]

Else:

[Module B]

[End of IF structure]

To be done by the student once...

Standard Notations and common functions

$\boxed{\text{Write a procedure:}}$ (class Assignment)

The solution of the quadratic equation

$$ax^2 + bx + c = 0$$

where $a \neq 0$, are given by the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

* The quantity $D = b^2 - 4ac$ is called discriminant of the equation. If D is negative then there are no real solution. If $D = 0$, then there only one (double) real solution, $x = -b/2a$.

* If D is positive, the formula gives the two distinct real solutions.