**Programming Methodology – I**

**1.Stages of Program Development:- (5 Stages)**

Defining a problem => Analysis Phase

Outlining the solution => Design Phase

Selecting & representing algorithms AND Coding => Coding Phase

4.Debugging, Testing & Validation => Debugging phase

Program Maintenance & Documentation => Maintenance Phase

Modular Programming – It refers to a technique which logical parts of a problem are divided into small self-explanatory, simple routines / procedures / functions so that each may be programmed independently.

The Advantages are:-

1. Change in application will not affect the whole program

2. Program understanding & maintenance is easy

3. Avoids confusion & makes programming logic easy

Characteristics of a good program:-

Accuracy

Clarity

Efficiency

Modularity

Reliability

User friendly

Portability – Ability of a program to run on different platform without doing much changes

Robustness – In ability of a program to handle unexpected inputs and hence errors.

**2.What does Stylistic guidelines do ?**

They provide an approach to make the presentation of a program look good, which can be done using: -

Use of comments – Improves readability & self-explanatory approach to programming

Blank Spaces / Blank Lines – provides clarity & readability

Proper Indentation

**Documentation**:-It is the process of collecting or storing information about a program on paper.There are two types of documentation

1. User Documentation

2. Technical Documentation

A Technical Documentation includes

I. Definition of problem

II. Need for program

III. Description of program

IV. Trouble Shooting

V. System specification required for executing the program

VI. User Manual – Explaining how to load program, use it & terminate program, expected output & required inputs.

Program Debugging :- Errors in a program are known as bugs & the process to remove the same is called as Debugging.

The Various errors are:-

Syntax Error

Run-time errors

Logical errors

Representational errors

Sematic errors

Syntax Error :- Error that arise when the rules of a language is violated, they are also known as complex errors as the compiler checks for syntax errors. For Example – missing a “;” at the end of the statement in C++

Run-time Error :- A runtime error is an error that causes abnormal program termination. i.e. it does not give errors during compilation but during execution for example :- C=a/b;

If “b” was given 0, it would result in known value resulting in divide by ZERO error.

Some Common Run-time errors :-

Divide by Zero

Logarithm of negative numbers

It is the responsibility of the programmer to take into account the error that could be encountered & give proper message to this effect.

Logical Error :- Error that is caused due to incorrect translators of either the problem or understanding of the problem

Representational Error :- Sometimes binary numbers cannot be represent decimal numbers accurately as there digits may get rounded off, this happens due to lack of storage bits hence precision of the no. is lost. This is known as representational error.

Sematic Error :- Errors that occurs when statements are not meaningful.

For Example : Y + Z = X;

5 = A;

**Programming Methodology – II**

**Problem Solving Methodology & Techniques :-**

I – Understanding the Problem :- Methods to analyse a problem step by step

a. Problem Definition

b. Requirement – analysis

c. Identification of operation

Problem Definition :- It defines what the problem is at this stage solution is not required to be provided. It is normally described from the user’s point of view and does not contain any technical words / terms.

Requirement – Analysis :- They form the first stage towards a solution. There are 3 major components;

1. Inputs (what inputs are required)

2. Process

3. Output (what final answer is expected)

Identification of operations :- In this stage we identify what type of operations are required for getting the output. The operations may be logical / mathematical

Example – with reference to C++

Relational - <,>,<+,>=,==

Logical – &&(and), || (or), !(not)

Control & looping – for, while, do while, go to

Process Checking :-

i. Dry run

ii. Independent inspection

iii. Structured Walk through

Dry Run – A Manual method of testing a process / program / algorithm for its correctness and functionality. In this method a table is created with different columns for each variable used in the program and the values of each variables is updated in the table as we proceed through program / algorithm.

Independent Inspection – in this method the specification and designed solution are first provided and then checked using dry run method.

Structured Walk-through – A team work, it involves presentation of a process or algorithm to a team of peers.

Problem Solving tools in the design phase (Stage two)

Outlining the Solution: -

Flow Chart

Algorithm

Pseudo Code

Data flow diagram

Source Code / Object Code :-

Program before compilation => source code

Program after compilation in machine language => object code