

## Unit-3

### PROGRAMMING METHODOLOGY

#### **Stylistic Guidelines:**

Writing good program is a skill. This can be developed by using the following guidelines .

1. Meaningful Names for identifiers: A programmer to give the meaningful names to each section of the program so that it can help him to identify the variable used for specific purpose. This helps him to execute the right elements during the complex run of a program.
2. Ensure clarity of expression: Expression carry out the specified action. Thus they must be clearly understood by the users. There should not be any compromise with the clarity of expression.
3. Use of comments and indentations: Comments generally are used as the internal documentation of a program . if comments are used in the program they will guide the program while debugging and checking. While indentation is the proper way of writing to avoid the confusion regarding the flow of program. These highlights nesting of groups of control statements.
4. Insert blank lines and blank spaces: Blank lines should be used to separate long, logically related blocks of code. Specifically Normally in programming the standard for the use of spaces is to follow normal English rules. This means that: Most basic symbols in C++ (e.g., "=", "+", etc.) should have at least one space before and one space after them.
5. Statements :Each statement should appear on a separate line. The opening brace following a control statement such as if or while should appear on the line after the if or while, lined up with the left of the control statement, and the closing brace should appear on its own line, lined up with the left of the control statement. As an example, see the for loop in Figure 1. The opening and closing braces for a function should be lined up in the same way. The statements within a { } pair are indented relative to the braces.

#### **Characteristics of a Good Program:**

Following are the characteristics of a good program.

1. Effective and efficient: The program produces correct results and is faster, taking into account the memory constraints.
2. User friendly: The program should be user friendly. the user should not be confused during the program execution . The user should get correct direction and alerts when he is going through the program.
3. Self documenting code: A good program must have self documenting code. This code will help the programmer to identify the part of the source code and clarify their meaning in the program.

4. Reliable: The good program should be able to cope up from any unexpected situations like wrong data or no data.
5. Portable: The program should be able to run on any platform, this property eases the use of program in different situations.

### **Stages of Program Development Process:**

A program development process is the step by step process in converting the inputs into outputs.

1. **Analysis**: this is the important phase where all the requirements of the program are gathered and the problem is cracked downed. An algorithm is formulated which gives the solution for the problem.
2. **Design** : In this phase of design a Model is developed which I look a like of a program . this phase gives the face to the program. Outputs are designed in this phase.
3. **Coding** : In this stage the algorithm is translated into the program called source code using some programming language.
4. **Compile the program**: Issue a compile command against source, and fix any compile errors that arise.
5. **Execute the program**: An error free program after compilation is put to run to produce the output. This phase is called run-time, the phase of program execution during which program instructions are carried out.

### **Robustness:**

Robustness is the ability of the program to bounce back an error and to continue operating within its environment.

**Documentation:** Documentation refers to written descriptions specification, design code and comments , internal and external to program which makes more readable and understandable.

Uses of documentation:

1. This becomes an useful interface between a technical personnel and non technical personnel.
2. This is very useful for upkeep and maintenance.
3. Documentation makes ease for any technical emergencies.
4. Very useful in operating for learners and trainers.
5. Enables trouble shooting when the application system breaks down.

## **PROBLEM SOLVING METHODOLOGY AND TECHNIQUES:**

To develop an efficient and effective programs we should adopt a proper problem solving methodology and use appropriate techniques. Following are some of the methods and techniques to develop a good program.

1. Understand the problem well : for a good program one should understand the problem well . one should know what exactly is expected from the problem. Knowing the problem well is the half way done.
2. Analyze the program. : analyzing the problem involves identifying the program specification and defining each program's minimum number of inputs required for output and processing components.
3. Code program : This step is the actual implementation of the program. In this program algorithm is translated into programming language. in this it is essential to decide which technique or logical will be more appropriate for coding.
4. Test and Debug program.: Once the solution algorithm is coded the next step is to test and debug the program. Testing is the process of finding errors in a program and debugging is of correcting the errors. The developed program is put to test in different conditions and verified at different level for its proper and efficient working.

\*\*\*\*\*