PRINCIPLES AND TECHNIQUES OF PROGRAMMING

SOFTWARE

An integrated set of programs which supplies instructionto a computer.

OR

Software is a set of instructions or mechanism which provides to the computer system , by this mechanism user interacted with the system .

OR

Software is a link between all its component to perform a better functionality.

Software is further classified into two categories

- **1). SYSTEM SOFTWARE** :- it controls the computer system and its resources like memory Input / Output devices. It also controls the disk drives and the micro-processor , schedule the work within the machine and enables the hardware to understand users command / Instructions. For example win 98, windows XP, win 2000 , win 2007, UNIX etc.

- **2).APPLICATION SOFTWARE**:- Application software is asset of one or more program or instruction designed to carry out operation for a specified application . The application software always runs based on any system software or operating system .
- In general we can say that it creates an interface / platform for the user to perform a specific type of job .
- For example :- Ms word for word processing
- Ms paint for draw an image
- Tally for accounting purpose

Relation between hardware and software

Hardware and software both are complimentary to each other , like buyer and seller (both are dependent on each other).

- Hardware cannot do anything itself without software and vice versa.

PROGRAM

- A program is a sequence of instructions that tell a computer how to do a task . When a computer follows the instructions in a program, we can say it *executes* the program.
- Computers are machines , and at the most basic level, they are a collection of switches where ‘1’ represents “on” and’0’ represents “off” .
- Everything that a computer does is implemented in this most basic of all numbering system is *Binary.*

BASIC GOALS OF COMPUTER PROGRAMMMING

- When you are planning to create a computer program you should:
- Ensure your program fulfills the need it is addressing.
- Ensure that people can easily use your program.
- Ensure that it is easy to understand ,fix ,and improve your program without a major time investment.
- Enhance problem solving using logic.
- Developing a high understanding of principles in programming.
- Demonstrate basic concepts , and software applications.

COMMON PROBLEMS

- These are very common mistakes programmer make:-
- Your program does not do a job better than an available alternative program. At best , you are re-inventing the wheel.
- Your program does not work as intended.
- Your program is too complicated or too primitive to be useful to most people.
- Other people, or yourself at a later time , can't understand the programming behind your program . this means your project won't grow.

SOLUTIONS

- 1.Utility and Usability
- 2. Maintainability

PROGRAM PROPERTIES

- Reliability
- Robustness
- Usability
- Portability
- Maintainability
- Efficiency / performance

COMMON PRINCIPLES OF PROGRAMMING

- No body in programming love to debug, maintain or make changes in complex code .”keep it simple” states that most systems work best if they are kept simple rather than making it complex, so when you are writing code your solution should not be complicated that takes a lot of time and effort to understand.
- “The purpose of programming is to reduce complexity, not to create it”.

COMMON PRINCIPLES OF PROGRAMMING

- Duplication of data ,logic or function in code not only makes your code lengthy but also wastes a lot of time when it comes to maintain ,debug or modify the code.

COMMON PRINCIPLES OF PROGRAMMING

- Your program can become larger and complex if you are writing some code which you may need in future but not at the moment. Most of the programmer while implementing software think about future possibility and add some code or logic for some other features which don’t need at present. We recommend all the programmers to avoid this mistake to save a lot of time and effort.

COMMON PRINCIPLES OF PROGRAMMING

- Separation of concern s principle partition a complicated application into different sections or domains. Each section or domain address a separate concern or has a specific job. Each section is independent of each other .

The two most important aspects of programming are;-

- Understand Program specification
- Design a logical solution.
- Program specifications mean the available input , the output requirements and the rules to be followed in getting at the out from the given.
- To understand the above specifications, we should design an appropriate solution to illustrate the logic of the program. For this purpose we use two tools, these are-

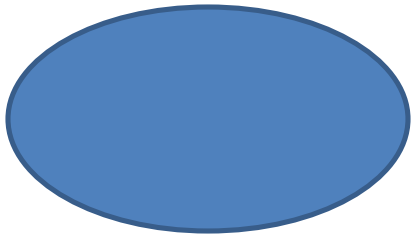
TOOLS

- **a).Flow Chart**
- **b). Pseudo code**
- Flow Charts are pictorial representation of the program logic, while Pseudo code describes the program logic in an informal English like language . Both are very versatile tools in logic development.

FLOW CHART

- Pictorial description of program logic
- Clarifies step by step logic of the program
- Symbolic representation of each Input,Output and Processing logic.
- Establishes a link between programmer and user
- Helps in understanding , error corrections ,and changing of programs.
- Independent of Program coding activity in any language like COBOL,C or PASCAL
- Part of program documentation

DIFFERENT SYMBOLS OF FLOW CHART



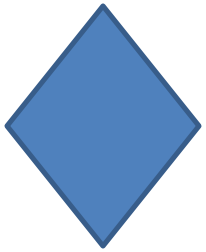
TERMINAL: Used for indicating start and stop of program logic.

- 

- **PROCESSING**:All steps in processing like arithmetic and logical operation.

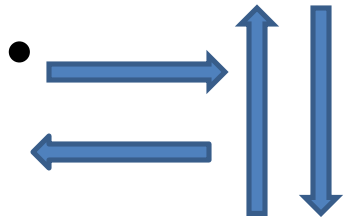
DIFFERENT SYMBOLS OF FLOW CHART

- **INPUT/OUTPUT:**-All input and output operations are written within.

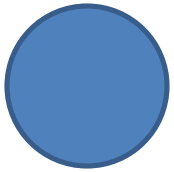


- **DECISION:**- Any operation which involves an answer to a question in the Form of yes or no are written in it

DIFFERENT SYMBOLS OF FLOW CHART



- **FLOW LINES**:The direction of the flow of CONNECTOR:-logic is drawn using arrow



- **CONNECTOR**:-used to connect logic flow lines .

- **SUBROUTINE:** commonly used routines are represented by this symbol.

PSEUDOCODE:-

- Logic Description in an informal English like language.
- Programming language independent description of logic
- Loose rules of grammer (syntax) .

S.U.COLLEGE ,HILSA (NALANDA)

- BCA/PT 1
- PAPER 2

RAKESH RANJAN
FACULTY (Dept.Of BCA)
S.U.COLLEGE ,HILSA (NALANDA)
PATLIPUTRA UNIVERSITY,PATNA