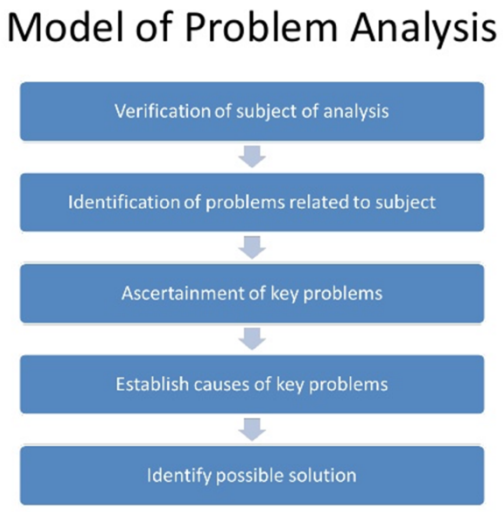
Unit-1

Problem Solving with computer

Problem Analysis

Problem analysis can be defined as dissecting and thoroughly studying a problem with the objective to understand how the problem emerged and how it grew to its current proportions.



Uses of Problem Analysis

- Performance Reengineering
- Decision Making
- Operation Management
- Benchmarking
- Competitive intelligence

Algorithm

An algorithm is the finite set of step by step instruction that solve a problem. An algorithm consists of stepwise list of instruction. The word algorithm comes from muslim mathematician, Abu Ja'far Mohammed ibn Musa al-Khowarizmi.

All algorithm should have the following properties.

1. It should have an input
2. The steps mentioned in an algorithm should be executable by the computer.
3. Each and every instruction should be in a simple language.
4. The number of steps should be finite.
5. It should not depend on a particular computer language or computer.
6. The algorithm should give an output after executing the finite number of steps.

Flowchart

Flowchart is the graphical representation of sequence of steps to be followed by a computer to solve a given problem.

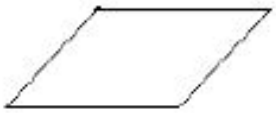
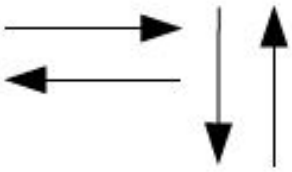
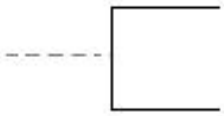
Advantages of flowchart

1. It is easier for a programmer to explain the logic of a program through flowchart.
2. It helps in effective analysis of the program.
3. Easy to detect, locate bugs in a program.
4. Coding becomes effective and faster.
5. Serve as documentation.

Disadvantages of flowchart

1. Very time consuming and laborious job.
2. Needs to remember symbols.
3. Difficult to correct error.

Symbols to draw flowchart

Symbol	Name	Meaning
	**Process Symbol**	Represents the process of executing a defined operation or groups of operations that results in a change in value, form, or location of information. Also functions as the default symbol when no other symbol is available.
	**Input/Output (I/O) Symbol**	Represents an I/O function, which makes data available for processing (input) or displaying (output)of processed information.
	**Flowline Symbol**	Represents the sequence of available information and executable operations.The lines connect other symbols, and the arrowheads are mandatory only for right-to-left and bottom-to-top flow.
	**Annotation Symbol**	Represents the addition of descriptive information, comments, or explanatory notes as clarification. The vertical line and the broken line may be placed on the left, as shown, or on the right.
	**Decision Symbol**	Represents a decision that determines which of a number of alternative paths is to be followed.
	**Terminal Symbol**	Represents the beginning, the end, or a point of interruption or delay in a program.

History of C programming language

- C was developed at Bell Laboratories in 1972 by Dennis Ritchie.
- Its principles and ideas were taken from the earlier language B and B's earlier ancestors BCPL and CPL.
- CPL (Combined Programming Language) was developed with the purpose of creating a language that was capable of both high level, machine independent programming and would still allow the programmer to control the behavior of individual bits of information.
- Drawback of CPL was that it was too large for use in many applications.
- In 1967, BCPL (Basic CPL) was created as a scaled down version of CPL.
- In 1970, Ken Thompson at Bell Labs developed B language.
- B was a scaled down version of BCPL written specifically for use in systems programming.

Features of C Language

1. Portability: we can easily transform a program written in C from one computer to another with few or no changes and compile with appropriate compiler.
2. Faster and efficient: C is faster and more efficient than comparable program. For eg a program to increment a variable from 0 to 15000 takes about 50 seconds in BASIC while it takes 1 second in C.
3. Supports structured Programming: Problem is solved in terms of function modules or blocks. The modular structures makes program debugging, testing and maintenance easier.
4. Extendibility: C is basically a collection of functions that are supported by C library. We can continuously add (or extend) our own functions to C library.
5. Flexible: It permits us to write any complex programs with the help of its rich set of in-built functions and operators. It is also called middle-level language because it permits us to write both application and system software.

Advantages of C language

- It is easier to interact with hardware.
- It is easy to learn.
- Efficient.

Disadvantage of C language

- Doesn't contain runtime checking.
- As the program extends it is very difficult to fix the bugs.

Structure of C program

Comments

Preprocessor directives

Global variables

main( ) function

{

local variables

statements

……

……

}

```
func1( )
{
    local variables
    statements
    ……….
    ……….
}
func2( )
{
    local variables
    statements
    ……….
    ……….
}
```

Coding

Once the design has been developed and reviewed, the next step is the actual writing of the program called coding. Coding is the process of translation from a detailed design to program statements in a programming language. The programmer writes the program (generates source code) using a language bounded by the rules of that language.

Compilation and execution

Source code of any high-level language must be translated into object code (machine level language) before it is executed. This translation process is known as compilation or interpretation, depending on how it is carried out. Finally the .exe file i.e executable file is produced and the program is executed.

Testing

Testing is the process of evaluating a newly developed program to check if it generates desired output or not. After debugging, the program may run fine in the laboratory. However, it then needs to be tested with real data.

It is compulsory to test with bad data (data that is faulty), or incomplete, or in large quantities to see if it can make the system crash or not.The testing process may involve several trials using different test data before the programming team is satisfied so that program can be released.

Debugging

Debugging is a process of detecting, locating and removing errors in a program. Mistakes may be syntax error or logical error or run- time error.

Syntax error

Syntax error is the error in the program's grammatical rule or in writing format. If a program contains syntax error, the program will not be executed and will not produce output. This error is detected by translator during translation. So, it is easier to correct. For example in C language whenever we do not use semicolon in a simple statement causes syntax error.

Logical error (semantic error)

A logical error is an error (bug) in the concept or logic of the program. It causes a program to operate incorrectly. If a program contains only a logical error, the program will be executed and will also produce output, but the output will not be accurate. For example, to calculate the simple interest the formula is i=(p*t*r)/100 but if we write i=(p*t*r)/10 then it will produce semantic error, because it is syntactically correct but if produce undesired output.

Run time error

When a program is running, or executing, it is said to be in run-time. A run time error is an error that is generated while the program is being executed. For example, a memory leak, where the program take up excessive amounts of system memory, then it may generate run-time error as it may allow other program to be loaded in the memory.

Documentation

Documentation is the process of collecting, organizing, storing and maintaining a complete record of system and other documents used or prepared during the different phases of the life cycle of system. Documentation is needed for everyone who will be involved with the program.

Documentation plays a vital role in maintenance of the program. Program maintenance is the process of updating software so that it continues to be useful. It is a costly process, but can be used to extend the life of a program. Documentation resulting from this step consists of the modified program package reflecting what problems occurred and what program changes were performed.

University Exam Questions

1. Write an algorithm and draw the flowchart to determine whether a given number is even or odd and explain it.[2068]
2. What an algorithm and flow chart to find out whether a given integer is zero, +ve or –ve and explain it. [2069]
3. What is logical error? Write flowchart and program for checking whether the number entered by the user is exactly divisible by 5 but not by 11. [2070]
4. What is an algorithm? Write an algorithm to check given number is prime or composite. [2070]
5. Explain the structure of C program with example. [2071]