Computer Programming:
Writing software (computer programs) involves describing processes, procedures; it involves the authoring of algorithms. Computer programming involves developing lists of instructions - the source code representation of software

In general, the programmer's job is to convert problem solutions into instructions for the computer. That is, the programmer prepares the instructions of a computer program and runs those instructions on the computer, tests the program to see if it is working properly, and makes corrections to the program. The programmer also writes a report on the program. These activities are all done for the purpose of helping a user fill a need, such as paying employees, billing customers, or admitting students to college.

The Programming Process:
Developing a program involves steps similar to any problem-solving task. There are five main ingredients in the programming process:

1. Defining the problem
2. Planning the solution
3. Coding the program
4. Testing the program
5. Documenting the program

Structured programming:
A technique for organizing and coding computer programs in which a hierarchy of modules is used, each having a single entry and a single exit point, and in which control is passed downward through the structure without unconditional branches to higher levels of the structure. Three types of control flow are used: sequential, test, and iteration.
Top Down Design is a method of designing something that starts with the complete item then breaks it down into smaller and smaller components. In programming it means breaking a difficult task down (divide and conquer) and solving pieces independently until every step can easily be implemented (successive refinement).

Removing goto from the programming language and replacing it with structures like for, while, and, if, is called structured programming. The for, while, and if structures represent the next level of abstraction up from assembly programming.

Structured programming was first suggested by Corrado Bohm and Guiseppe Jacopini. The two mathematicians demonstrated that any computer program can be written with just three structures: decisions, sequences, and loops. Edsger Dijkstra's subsequent article, Go To Statement Considered Harmful was instrumental in the trend towards structured programming. The most common methodology employed was developed by Dijkstra. In this model (which is

often considered to be synonymous with structured programming, although other models exist) the developer separates programs into subsections that each have only one point of access and one point of exit.

Almost any language can use structured programming techniques to avoid common pitfalls of unstructured languages. Unstructured programming must rely upon the discipline of the developer to avoid structural problems, and as a consequence may result in poorly organized programs. Most modern procedural languages include features that encourage structured programming. Object-oriented programming (OOP) can be thought of as a type of structured programming, uses structured programming techniques for program flow, and adds more structure for data to the model.

Advantages:

- Easy to write:
  Modular design increases the programmer's productivity by allowing them to look at the big picture first and focus on details later.

  Several Programmers can work on a single, large program, each working on a different module

  Studies show structured programs take less time to write than standard programs.

  Procedures written for one program can be reused in other programs requiring the same task. A procedure that can be used in many programs is said to be reusable

- Easy to debug
  Since each procedure is specialized to perform just one task, a procedure can be checked individually. Older unstructured programs consist of a sequence of instructions that are not grouped for specific tasks. The logic of such programs is cluttered with details and therefore difficult to follow.

  Easy to Understand
  The relationship between the procedures shows the modular design of the program.

  Meaningful procedure names and clear documentation identify the task performed by each module
  Meaningful variable names help the programmer identify the purpose of each variable.

  Easy to Change
  Since a correctly written structured program is self-documenting, it can be easily understood by another programmer