

Structured Programming

Structured programming (sometimes known as *modular programming*) is a programming paradigm that facilitates the creation of programs with readable code and reusable components. All modern programming languages support structured programming, but the mechanisms of support, like the syntax of the programming languages, varies.

Structured programming encourages dividing an application program into a hierarchy of modules or autonomous elements, which may, in turn, contain other such elements. Within each element, code may be further structured using blocks of related logic designed to improve readability and maintainability.

Modular programming, which is today seen as synonymous with structured programming, emerged a decade later as it became clear that reuse of common code could improve developer productivity. In modular programming, a program is divided into semi-independent modules, each of which is called when needed. C is called structured programming language because a program in c language can be divided into small logical functional modules or structures with the help of function procedure.

Advantages of structured programming

The primary advantages of structured programming are:

1. It encourages top-down implementation, which improves both readability and maintainability of code.
2. It promotes code reuse, since even internal modules can be extracted and made independent, residents in libraries, described in directories and referenced by many other applications.
3. It's widely agreed that development time and code quality are improved through structured programming.
4. It is user friendly and easy to understand.
5. Similar to English vocabulary of words and symbols.
6. It is easier to learn.
7. They require less time to write.
8. They are easier to maintain.
9. These are mainly problem oriented rather than machine based.
10. Program written in a higher level language can be translated into many machine languages and therefore can run on any computer for which there exists an appropriate translator.
11. It is independent of machine on which it is used i.e. programs developed in high level languages can be run on any computer.

Disadvantages of structured programming

The following are the disadvantages of structured programming:

1. A high level language has to be translated into the machine language by translator and thus a price in computer time is paid.
2. The object code generated by a translator might be inefficient compared to an equivalent assembly language program.
3. Data types are proceeds in many functions in a structured program. When changes occur in those data types, the corresponding change must be made to every location that acts on those data types within the program. This is really a very time consuming task if the program is very large.
4. Let us consider the case of software development in which several programmers work as a team on an application. In a structured program, each programmer is assigned to build a specific set of functions and data types. Since different programmers handle separate functions that have mutually shared data type. Other programmers in the team must reflect the changes in data types done by the programmer in data type handled. Otherwise, it requires rewriting several functions.