Common Programming Errors

Errors like division-by-zero or array bound errors occur as a program runs, so these errors are called as run-time errors or execution time errors. These are generally fatal errors i.e. an error that causes the program to terminate immediately without having successfully performed its job. Non fatal errors allow programmers to run to completion, often producing incorrect errors.

Forgetting to include the *iostream.h* file in a program that inputs data from the keyboard or outputs data to the screen causes the compiler to issue an error message is very common among beginners particularly if you are not using an IDE.

Omitting the semicolon at the end of a statement is a syntax error. A syntax error is caused when the compiler cannot recognize a statement. The compiler normally issues an error message to help the programmer locate and fix the incorrect statement. Syntax errors are violation of the language. Syntax errors are called compiler errors, compile time errors, compilation errors because they appear during compilation phase.

Good Programming Practices

Write your C++ program in a simple and straightforward manner. This is sometimes referred to as KIS (“Keep it Simple”). Don’t stretch the language by trying bizarre usages

If you are working professional, then read the manuals for version of C++ you are using. Refer to these manuals frequently to be sure you are aware of the rich collection of C++ features and that you are using these features correctly.

Your computer and compiler are good teachers. If after carefully viewing and listening videos and reading C++ manuals, you are not sure how a feature of C++ works, experiment using a small “*test program”* and see what happens. Set your compiler options for “*maximum warnings*”. Study each message you get when you compile your program and correct the program to eliminate the message.

Every program should begin with a comment describing the purpose of the program.