1. “Stack overflow”

Stack overflow means stack ran out of memory. The most common cause of stack overflow error is excessively deep or infinite recursion.

1. “Undefined reference to main”

It means compiler cannot find main function. It happens when you have many .cpp files, and you want to compile them separately without linking to the main.cpp. You can either use “g++ -c file.cpp”, or you must link to the main.cpp when you compile.

1. “`fooVar' undeclared (first use this function)”

The compiler does not know what “fooVar” is. It happens when you forget to include the header file for this function. You misspelled the name of that function. You misspelled header file name. Or “fooVar” function does not exist at all.

1. foo.h: No such file or directory

The compile cannot find foo.h in current directory. This may be because you haven’t created the header file yet, or that you either misspelled the header file name in the #include directive, or that you made a mistake in naming the header file when you created it with your editor.

1. “unterminated string or character constant”

It basically means you are missing a closing quote, and it happens when you forget to end literal strings with the closing quote.

1. “ANSI C++ forbids comparison between pointer and integer”

It means you are comparing a pointer to an integer, which is now allowed. It happens when you use double quotes to quote a character and you compare it to another character

(ie: char MyChar;

“A” == MyChar;)

If you want to denote a single character, use the single tick marks, as in 'A'.

1. “Null pointer exception” or “Segmentation Fault”

It means that you try to access memory that you don’t have permission to access. There are four common mistakes that lead to segmentation faults: dereferencing NULL, dereferencing an uninitialized pointer, dereferencing a pointer that has been deleted or that has gone out of scope (in the case of arrays declared in functions), and writing off the end of an array.

1. `cout' undeclared (first use this function)

It means the compiler does not know what “cout” is. This happens when you did not type “using namespace std;” or you did not type “std::cout”. Either way is the solution for this problem.

1. parse error before `something'

They often mean that the line of code above that is missing a semicolon.

1. “'member' is not accessible”

This means that you do not have the permission to access the member you try to access. This happens when you try to access the private members of a class from a different class, or when you try to access the protected members from a not inherited class.

1. “Destructor name must match the class name”

Like constructors, destructors must match the class name. This may happen when you misspell the destructor name or the cases are not the same.

1. “Friends must be functions or classes”

Friends cannot be variables, and they must be either functions or classes. This may happen when you type the keyword “friend” in front of something variables, which are not classes or functions