* C++ is less efficient than C because it involves object creation overhead along with function calling mechanism which increases running time of program.

Ffici y **Efficiency**

* C++ needs extra ROM (read only memory) space than C.
* C is faster than C++ because it involves only primitive function calling.
* C++ template, STL and C++ are much similar in efficiency because STL stack also uses templates.

**Ease of Implementation**

* C++ is much easier in terms of implementation because of its Oops feature.
* C++ is much similar to C++ template because whole code is same except data

Type parameterization in template.

* STL is even More easier to use because we have tom use only predefined functions and we can enjoy power of Object Oriented Programming.

**Testability**

* C is difficult to test in general.
* C++ is easier to test than C because we can test each class separately.
* C++ templates are easier than C++ because different data types can be checked in a single implementation.
* STL is easiest to test because all library functions are checked already.
* C is very hard to maintain and least robust because C does not Classes.

**Robustness and maintainability**

* C++ is well arranged using Classes and their methods hence easier to maintain.
* C++ template are also well arranged and more robust because of type parameterization.
* STL is very much easier to maintain because library function are well written already by other programmers and hence much easier to maintain.

**Readability**

* C code are least readable.
* C++ is better in terms of readability because of division of program in classes.
* C++ is better than C++ because by seeing definition of variables we can get ideas of its primitive elements for example stack<int> indicates stack of integers.
* STL is best in terms of readability because of library functions and templates.