**THREAD SAFETY**

An object is thread-safe for reading from multiple threads. For example, given an object A, it is safe to read A from thread 1 and from thread 2 simultaneously.

If an object is being written to by one thread, then all reads and writes to that object on the same or other threads must be protected. For example, given an object A, if thread 1 is writing to A, then thread 2 must be prevented from reading from or writing to A.

Refer this: <https://stackoverflow.com/questions/12931787/c11-stl-containers-and-thread-safety>

STL containers offer the following basic thread-safety guarantee:

1. simultaneous reads of the same object are OK
2. simultaneous read/writes of different objects are OK

But we have to use some form of custom synchronization (e.g. critical section) if you want to do something different, like e.g. simultaneous writes on the same object.