* Don’t make statements about efficiency of code without first doing time measurements. Guesses about performance are most unreliable.
* **Advice –**
* Use resource handles to manage resources (RAII).
* Use *unique\_ptr* to refer to objects of polymorphic type.
* Use *shared\_ptr* to refer to shared objects.
* Use type-safe mechanisms for concurrency.
* Minimise the use of shared data.
* Don’t choose shared data for communication because of “efficiency” without thought and preferably not without measurement.
* Think in terms of concurrent tasks, rather than threads.
* A library doesn’t have to be large or complicated to be useful.
* Time your programs before making claims about efficiency.
* You can write code to explicitly depend on properties of types.
* Use regular expressions for simple pattern matching.
* Don’t try to do serious numeric computation using only the language; use libraries.
* Properties of numeric types are accessible through numeric\_limits.