Introduction

This library gives us the power to run multithreads based on our time requirements.

WE'LL COVER THE FOLLOWING ^

- Time point:
- Time duration:
- Clock:

The time library is a key component of the new multithreading capabilities of C++. So you can put the current thread by

```
std::this_thread::sleep_for(std::chrono::milliseconds(15) for 15
milliseconds to sleep, or you try to acquire a lock for 2 minutes:
lock.try_lock_until(now + std::chrono::minutes(2)). Beside that, the chrono library makes it easy to perform simple performance tests:
```

```
// performanceMeasurement.cpp
#include <iostream>
#include <vector>
#include <chrono>
using namespace std;

int main(){
   std::vector<int> myBigVec(10000000, 2011);
   std::vector<int> myEmptyVec1;

auto begin= std::chrono::high_resolution_clock::now();
   myEmptyVec1 = myBigVec;
   auto end= std::chrono::high_resolution_clock::now() - begin;

auto timeInSeconds = std::chrono::duration<double>(end).count();
   std::cout << timeInSeconds << std::endl; // 0.0150688800 <- may vary from execution to execute return 0;
}</pre>
```







The time library consists of the three components, time point, time duration and clock.

Time point: #

Time point is defined by a starting point, the so-called epoch, and an additional time duration.

Time duration: #

Time duration is the difference between two time-points. It is given by the number of ticks.

Clock: #

A clock consists of a starting point (epoch) and a tick, so that the current time point can be calculated.