Exercise #1: a simple timer class

The goal of this first exercise is to let you familiarize with the CSCS system environment and to understand how to measure the amount of time/cycles spent by some specific portions of code. To achieve this goal, you’re asked to download and compile a ready-made class providing a simple timer implementation.

First, log into todi.cscs.ch by using your CSCS account. You can use ssh –X [username@ela.cscs.ch](mailto:username@ela.cscs.ch) and then ssh –Y [username@todi.cscs.ch](mailto:username@todi.cscs.ch) (-Y works as –X).

For this first exercise you don’t need to set any special module or to allocate resources.

Download or copy to your home the content of the “CscsTimer” directory (from the exercises section of the Autumn School portal).

There are three main project files:

* cscs\_timer.cpp and cscs\_timer.h are a stand-alone static class for measuring time and CPU cycles
* main.cpp is an example on how to use the CscsTimer class

The code is Linux/Windows compatible and, just in case of, a Visual Studio 2012 solution is also provided.

Compile and run the example. What is the CPU frequency? (Make an educated guess…)

Have a look at the code (e.g., using *gedit*). Add a method for computing the Fibonacci sequence. Modify the code in order to measure the time required to compute each of the first N numbers of the Fibonacci sequence. Suggestion: don’t use anything bigger than N=44.