Description(Bilal Atım)

I write 4 cpp file.

Thread0.cpp : I don't create any pthread. I create 10 function and call them in the main. Functions return void. I define variables like mean, sum, etc. and change these variables in the functions. I calculate time before calling functions and after functions. My unit of time is **microsecond**. I wrote the results to the **output1.txt** file.

Thread5.cpp : I create 5 pthread in the main. I create 5 functions for pthreads. Each pthread function call 2 function. I grouped functions by 2. I calculate time before creating threads and after joining pthreads. My unit of time is **microsecond**. I wrote the results to the **output2.txt** file.

Thread10.cpp : I create 10 pthread in the main. I create 10 functions for pthreads. Each pthread function call 1 function (pthread_func call inside sum_function). I calculate time before creating pthreads and after joining pthreads. My unit of time is **microsecond** in the last line of output. I wrote the results to the **output3.txt** file.

Thread_opt.cpp : I get pthreads number from second argument. If there is no second argument, my cpp file don't create any threads and work like Thread0.cpp file. If there is a second argument, In for loop I create pthreads according to number of input. Each pthread goes to same function. In that function, I set a range for each pthread. Each pthread calls the function in its range. It doesn't matter if the number of threads is 1,3,4,5 or 8 each thread runs its own functions. My unit of time is **microsecond** in the last line of output. I wrote the results to the **output4.txt** file.

My Measurements(in VM):

4997 microseconds without pthreads | 4659 microseconds with 5 pthreads

6117 microseconds with 10 pthreads | 6252 microseconds with 10 pthreads in Thread_opt.cpp

Looking at these two results, using threads speeds up the program. This is because threads are running simultaneously at the same time. But when we look at the 10 threads case, it is slower then without threads case. There may be many reasons for this. Since it is run at different times, the running time of the program also depends on variables such as the state of the computer while it is running. Also, the time it takes to generate threads 10 times may have affected the runtime too much. In addition to this, I measure after all threads join the main thread. We add to the time that all treads join the main thread. If I calculate time before joining measurement with 10 threads will be 10 times lower, then without threads. Also, some functions take more time. These functions determine the working time. If we worked on 10 different functions in the same runtime and the execution time of each function will be too much than the thread creation time. We can easily see that when the number of threads will be x times, working time becomes 1/x times because each thread work at the same time.