T.C. GAZİ UNIVERSITY

ENGINEERING FACULTY

COMPUTER ENGINEERING DEPARTMENT

BM479E – PARALLEL COMPUTING ARCHITECTURES AND PROGRAMMING

ASSIGNMENT– 1: CALCULATING THE PI NUMBER USING MONTE CARLO METHOD AND MPI LIBRARY

MUHAMMED EMRE EMRAH

131180027

OCTOBER 2017

1. Introduction

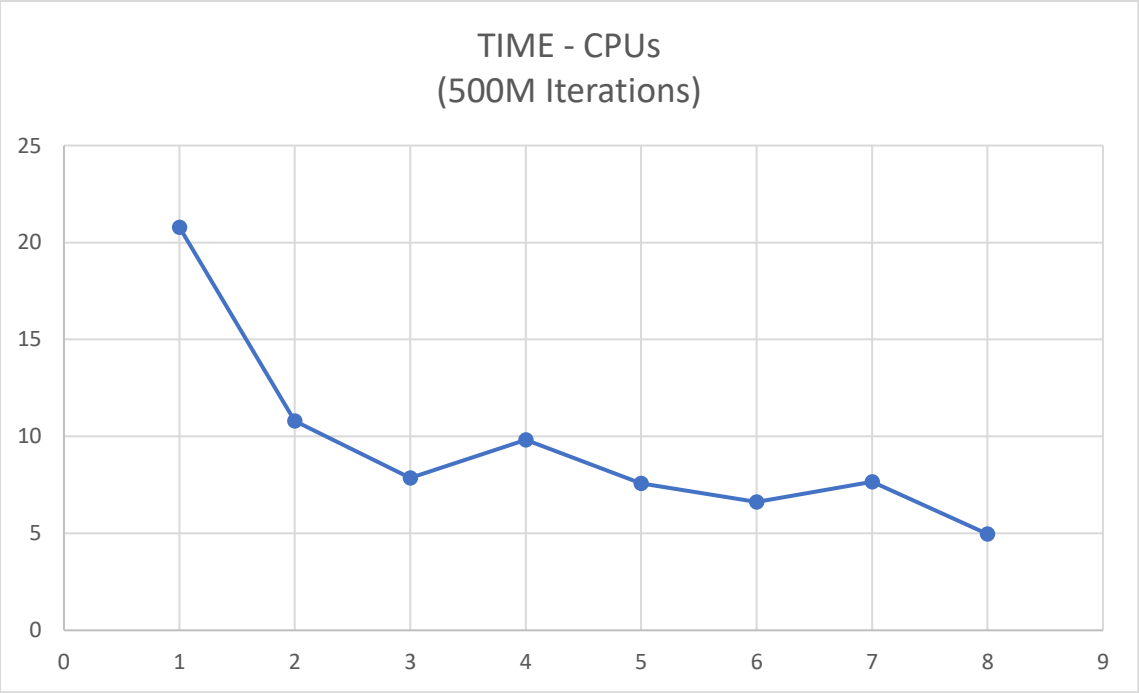
In this assignment, we are going to compute the Pi number up to a point with Monte Carlo method, using the MPI library. Then, we will get some results and show them in a bunch of charts.

2. Code

Given in the appendix. All explainations written as comment in code.

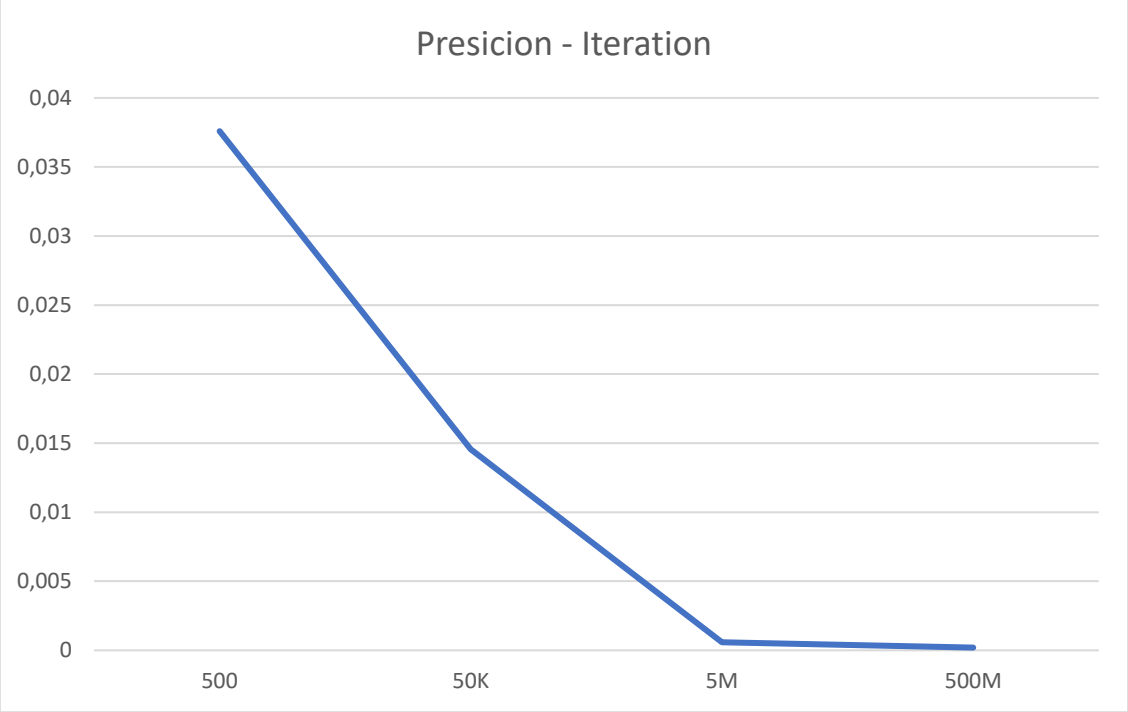
3. Graphics and Analysis

In Time (seconds) – CPU (amount) chart for 500M iterations (500 million darts), we clearly see that by the amount of CPUs is rising, the amount of time to compute 500M iterations is clearly drops. The drop rate is much higher at low amount of CPUs, it's nearly 2x faster compared to 2 CPUs to 1 CPU; but it's not when comparing to 4 CPUs to 8 CPUs:



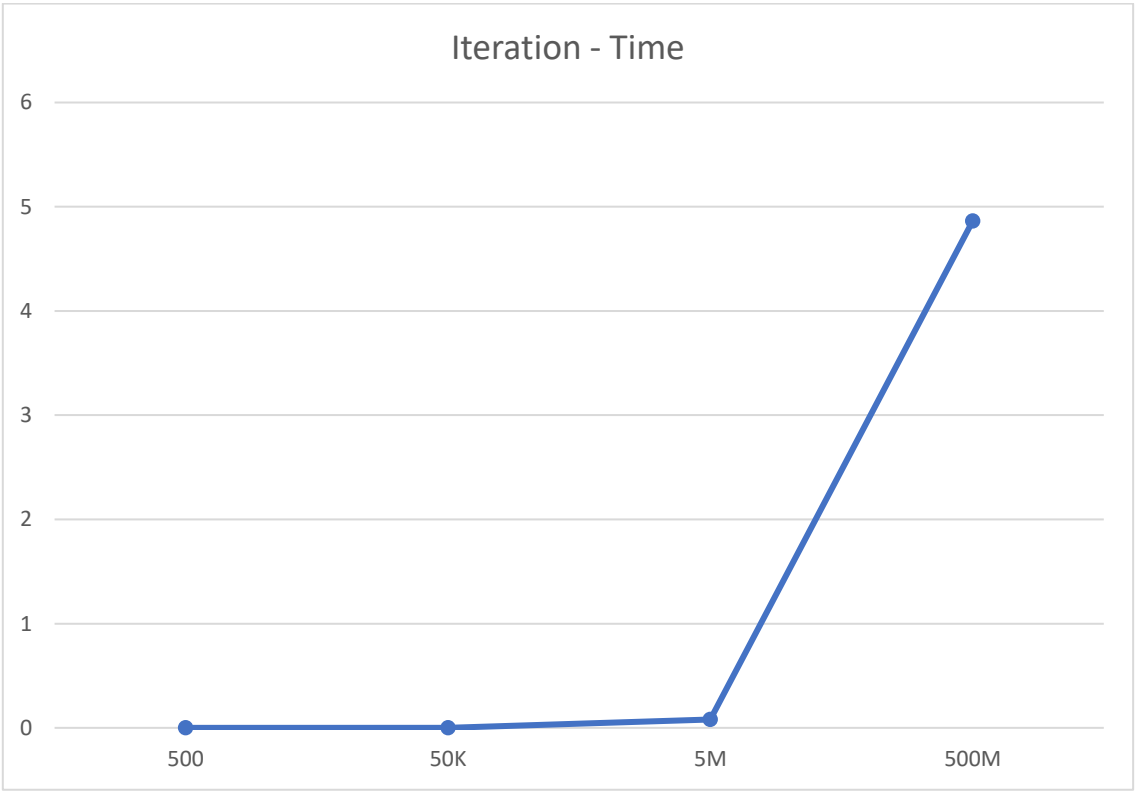
1.1. TIME-CPU chart FOR 500M darts

And here is Precision – Iteration chart. It’s clear that the higher amount of iterations we made, the higher precision (lower difference between actual Pi and our computed Pi number) we have:



1.2. Presicion – Iteration chart for 4 CPUs

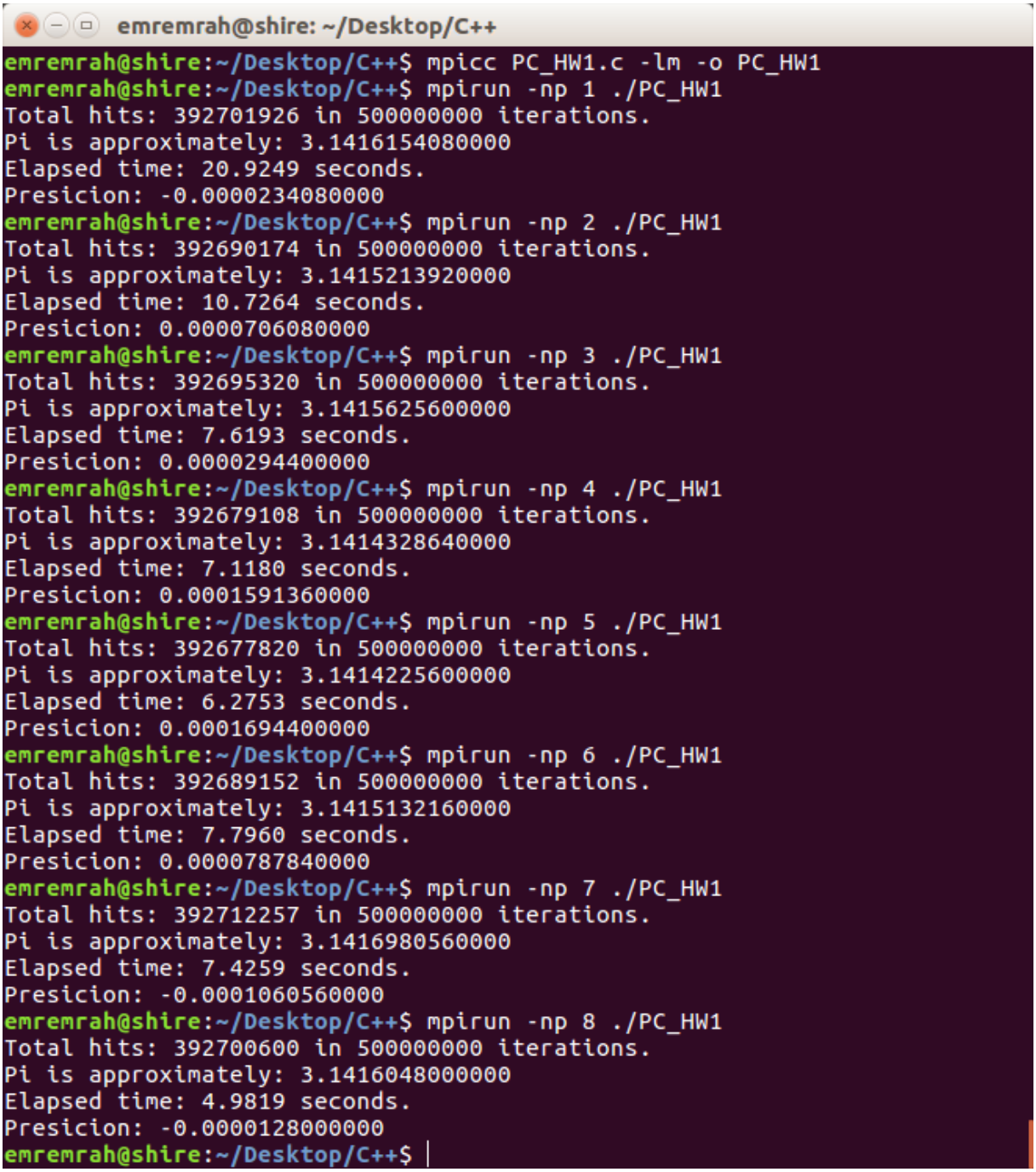
And for last, here is Time – Iteration chart for 8 CPUs.



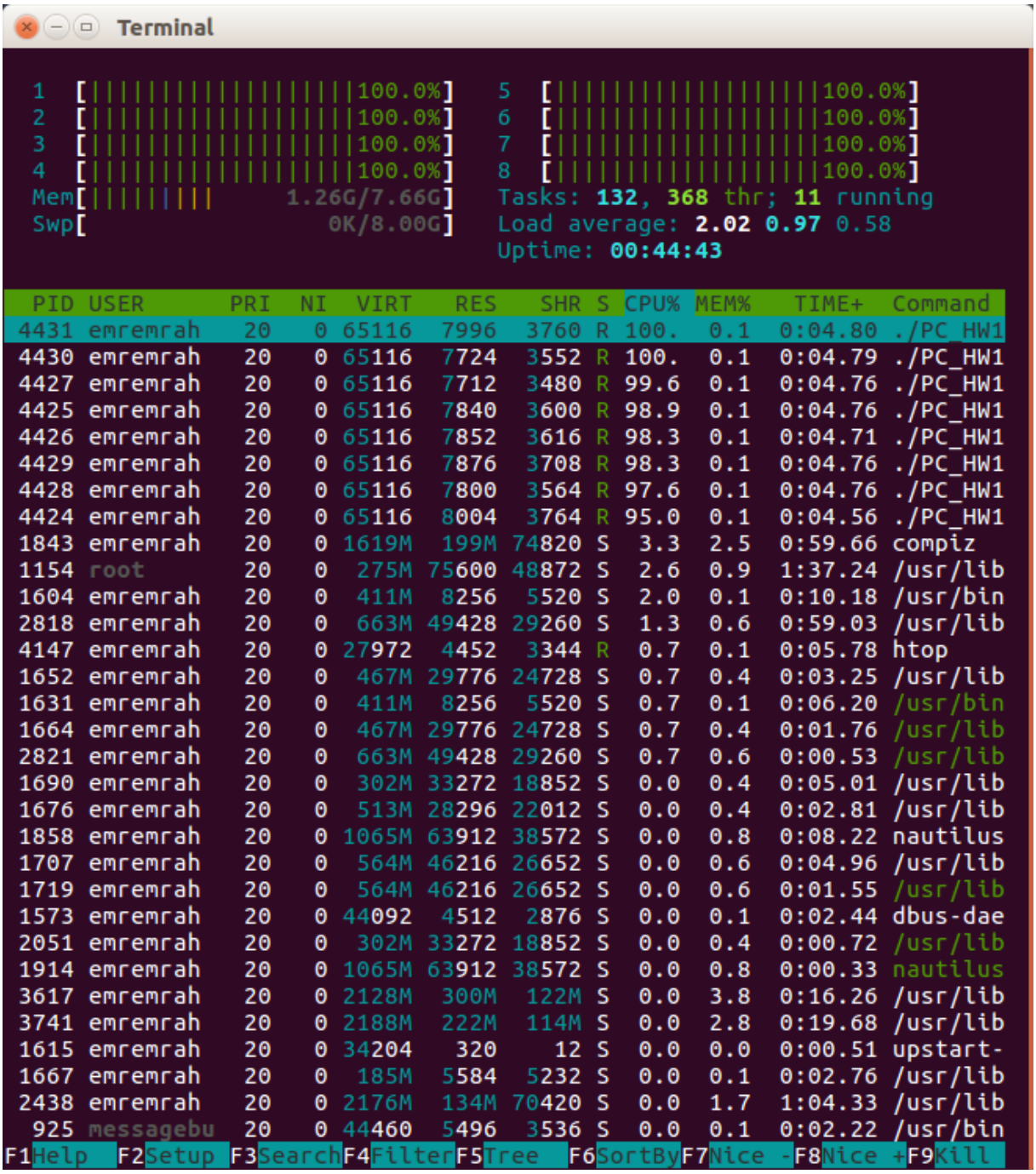
1.3. Iteration – Time (seconds) chart for 8 CPUs.

4. APPENDIX

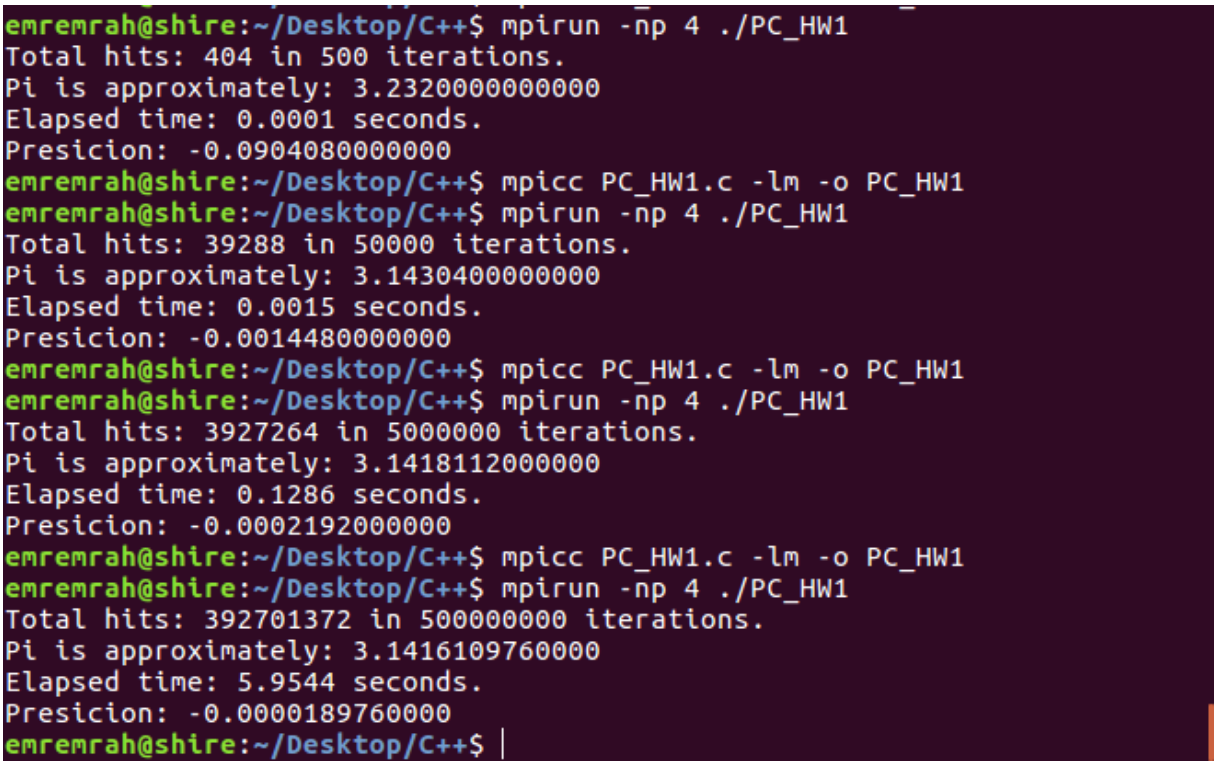
4.1. Screenshots for HOTP (task manager).



4.1.1. Linux terminal when we run our code from 1 to 8 CPUs increasing one-by-one



4.1.2. Linux terminal when computing 500M iterations with 8 CPUs



4.1.3. Linux terminal when we run 8 CPUs to compute 500 to 500M iterations

4.2. My Code

Given in next page.