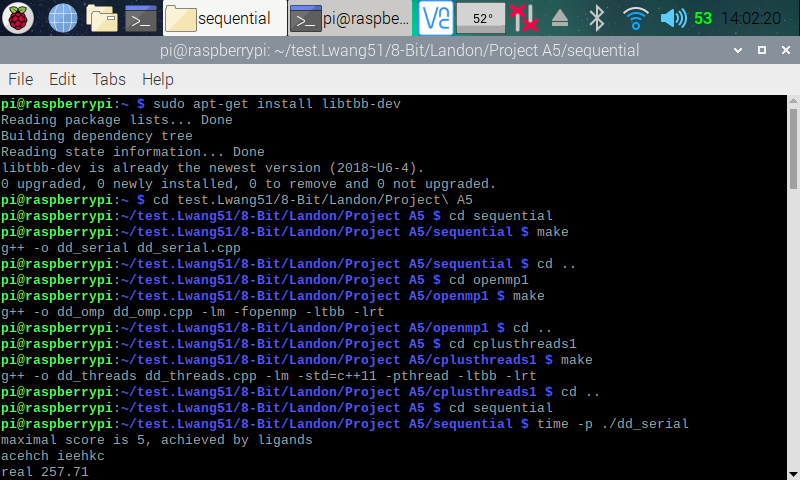
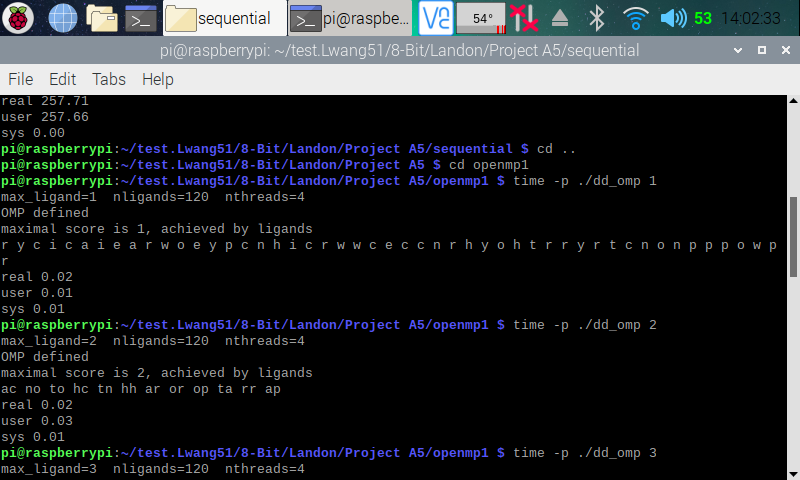
**Parallel Programming A5**

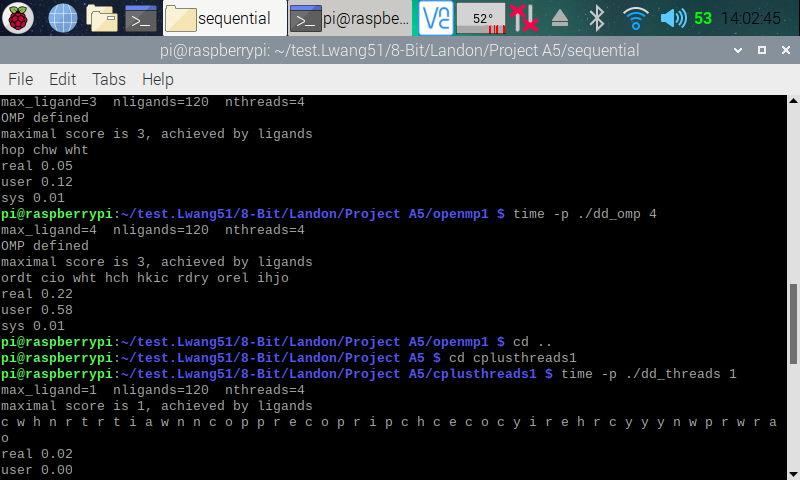
**Drug Design and DNA in Parallel**

**Part 1: Sequential, OpenMP, and C++ 11 Threads Solutions and Measuring Run Time**

****

Here, (in the screenshot above), I ran into an error while compiling the OpenMP and C++ 11 Threads program. The error stated “*tbb/concurrent\_vector.h: no such file or directory”*. After doing some research and looking through my teammate’s explanation, I found out that I had to install the TBB library onto my Raspberry Pi. Using the commands “*sudo apt-get install libtbb-dev”*, I installed the library then compiled the OpenMP and C++ 11 Threads solution programs.







Here (in the three screenshots above), I used the command *time -p ./ProgramName* to find the runtime of each program. I also tested the runtime for the two parallel solution programs with 2, 3, and 4 threads. We can see that the runtime increase as we increase the amount of threads.

**Part 2: Discussion Questions**

|  |  |
| --- | --- |
| Implementation | Times (s) |
| dd\_serial | 257.71 |
| dd\_omp | 0.02 |
| dd\_threads | 0.02 |

|  |  |  |  |
| --- | --- | --- | --- |
| Implementation | Time (s) 2 Threads | Time (s) 3 Threads | Time (s) 4 Threads |
| dd\_omp | 0.02 | 0.05 | 0.22 |
| dd\_threads | 0.02 | 0.04 | 0.19 |

1. **Which approach is the fastest?**

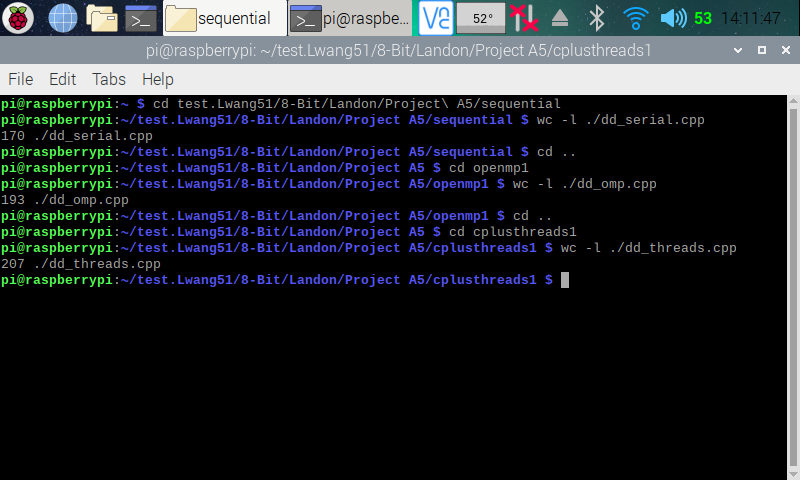
Dd\_threads implementation was faster than the other two solutions

1. **Determine the number of lines in each file (use wc -l). How does the C++11 implementation compare to the OpenMP implementations?**

Sequential Program: 170 lines

OpenMP Program: 193 lines

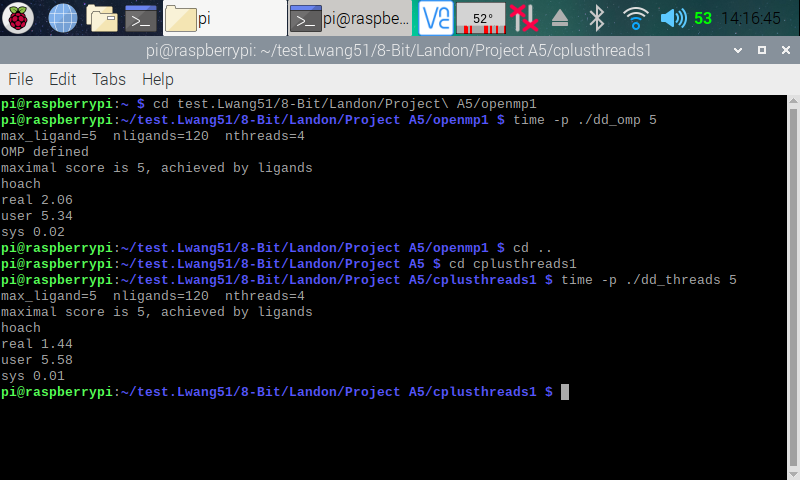
C++ 11 Threads Program: 207 lines



1. **Increase the number of threads to 5 threads. What is the run time for each?**

OpenMP Program: 2.06 seconds

C++ 11 Threads Program: 1.44 seconds



1. **Increase the maximum ligand length to 7, and rerun each program. What is the run**

**time for each?**

OpenMP Program: 152.57 seconds

C++ 11 Threads Program: 081.61 seconds

