**A5 Parallel Programming – Team Hustle**

We have encountered the following error when trying to compile the OpenMP solution and C++ 11 threads solution:

/usr/local/include/tbb/machine/gcc\_armv7.h:31:2 error: #error compilation  
requires an ARMv7-a architecture.

To resolve the error, we followed the steps in these three websites:

1. <https://www.pyimagesearch.com/2018/09/19/pip-install-opencv/>

* To pip install OpenCV on Raspberry Pi

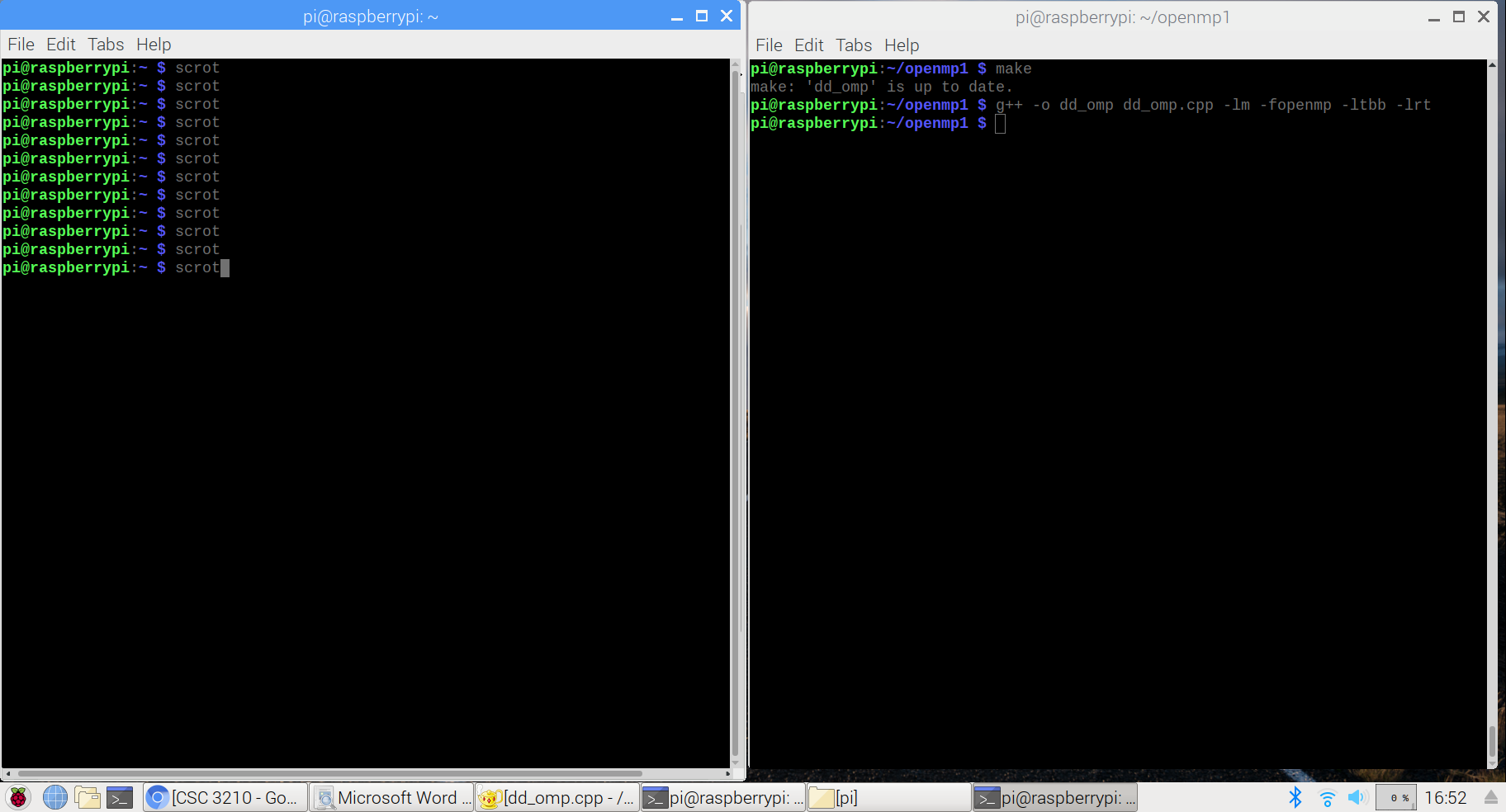
1. <https://www.theimpossiblecode.com/blog/intel-tbb-on-raspberry-pi/>

* To build a TBB package for Raspberry Pi

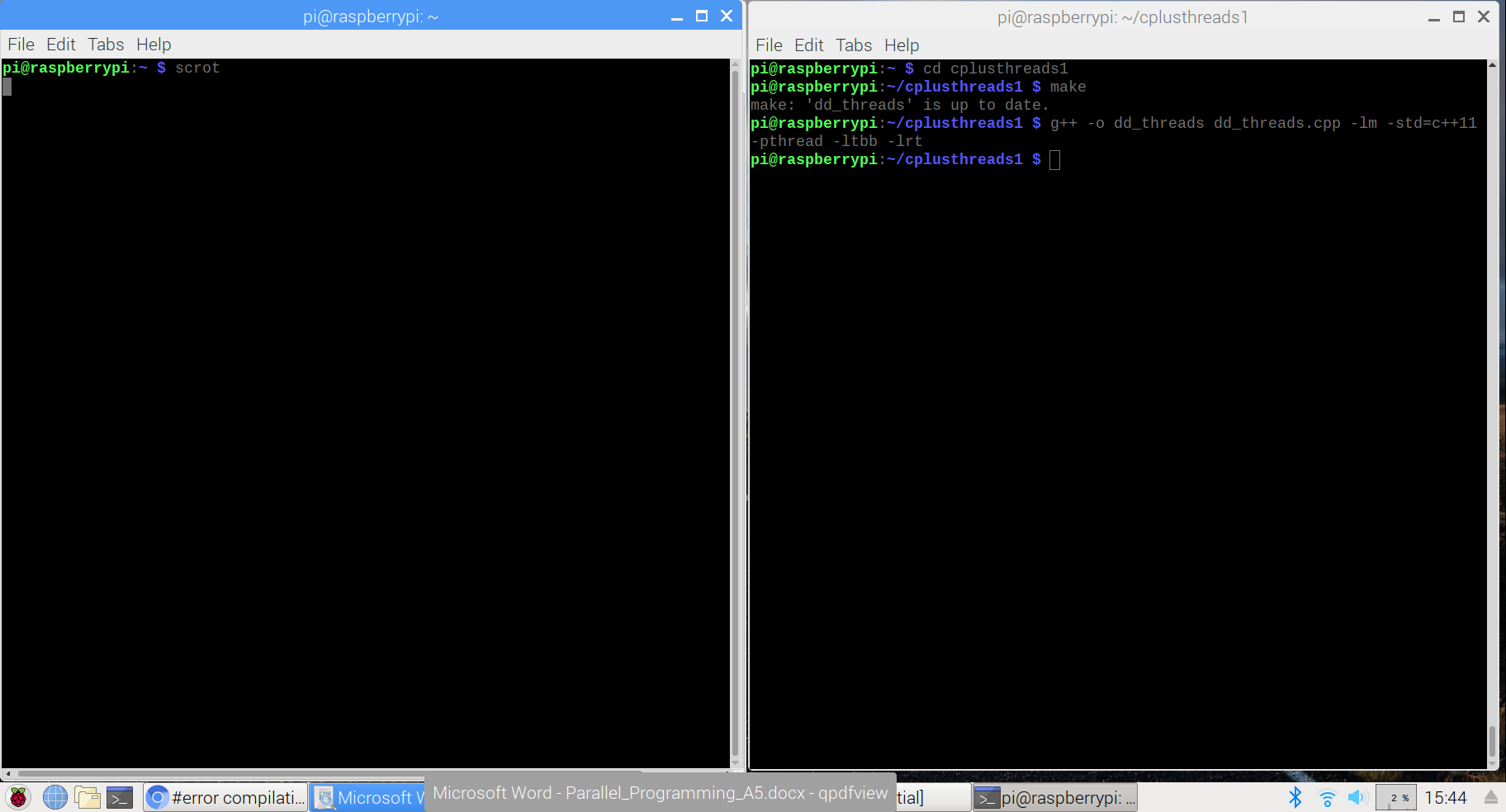
1. <https://stackoverflow.com/questions/30131032/compile-opencv-with-tbb-on-raspberry-pi-2>

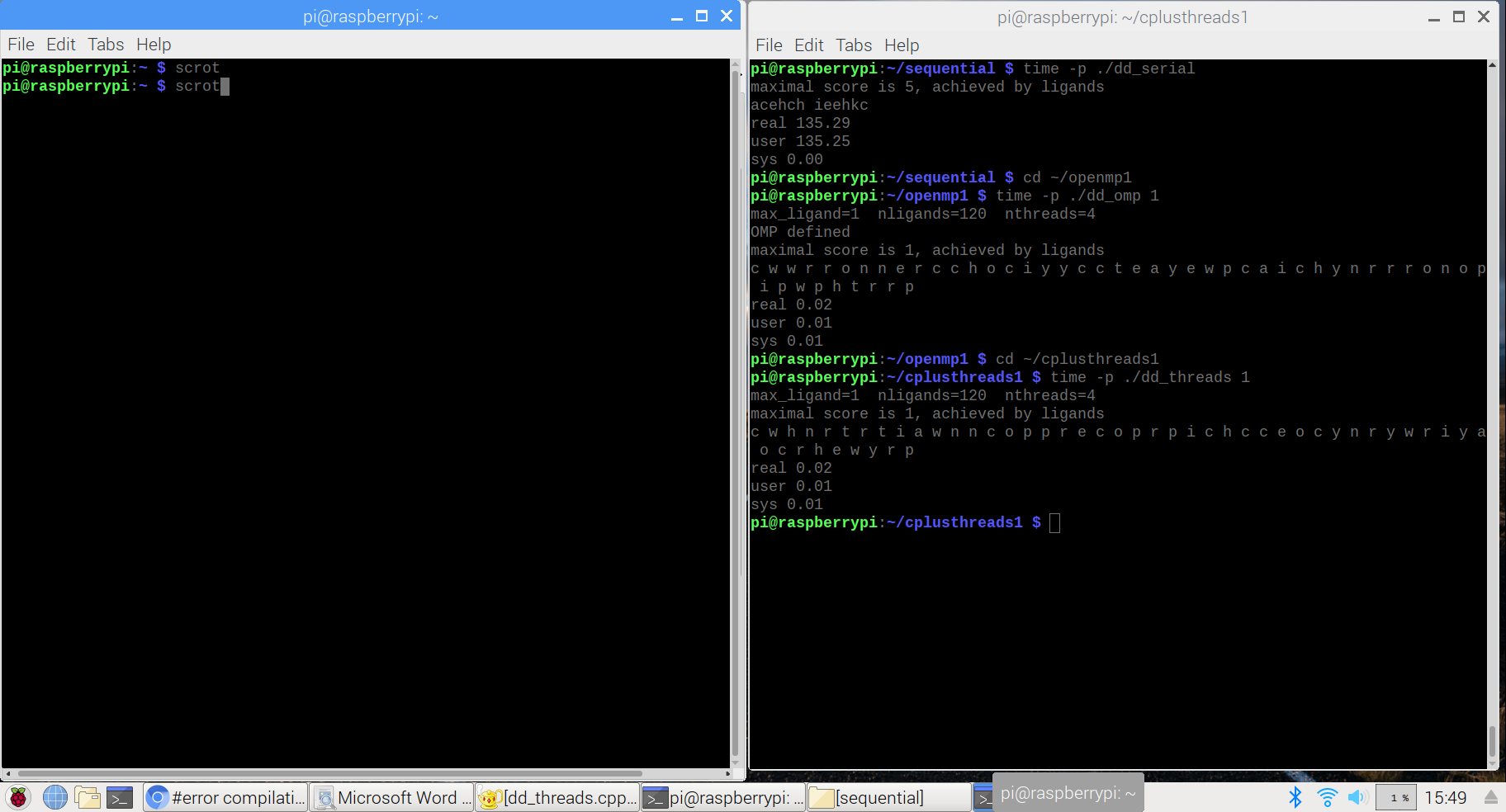
* To change and replace some codes inside the gcc\_armv7.h file

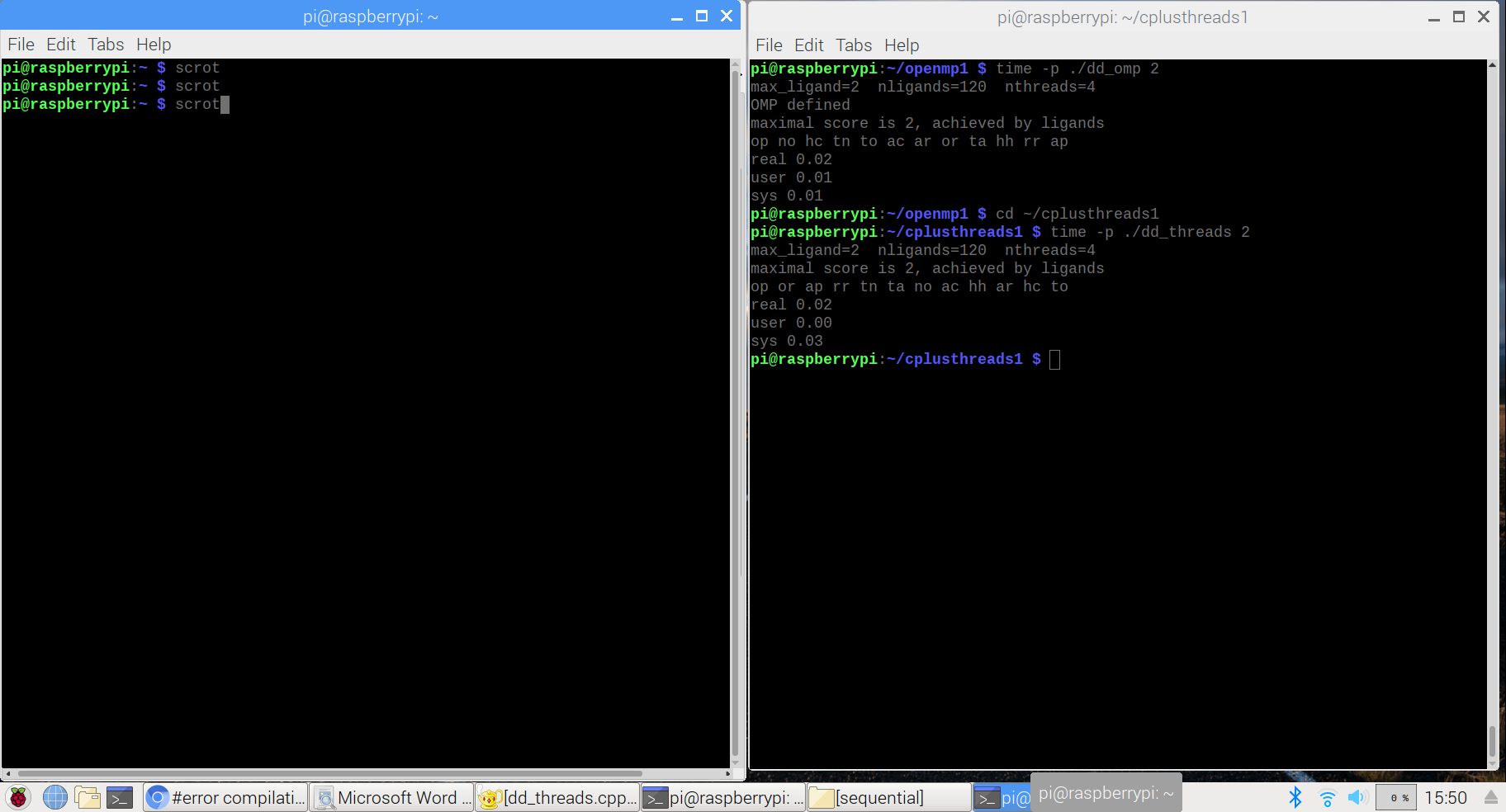
3. OpenMP Solution Compilation

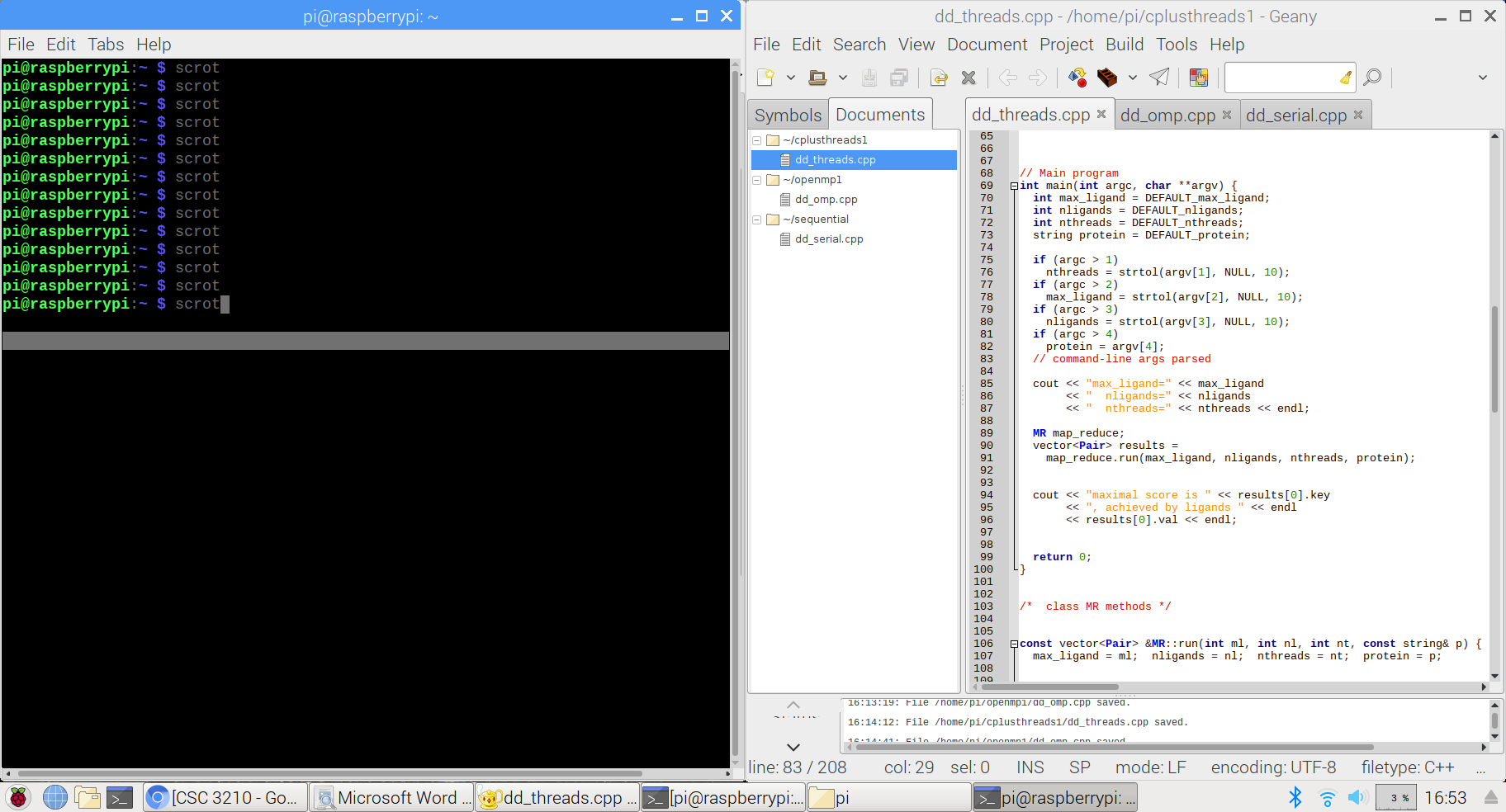
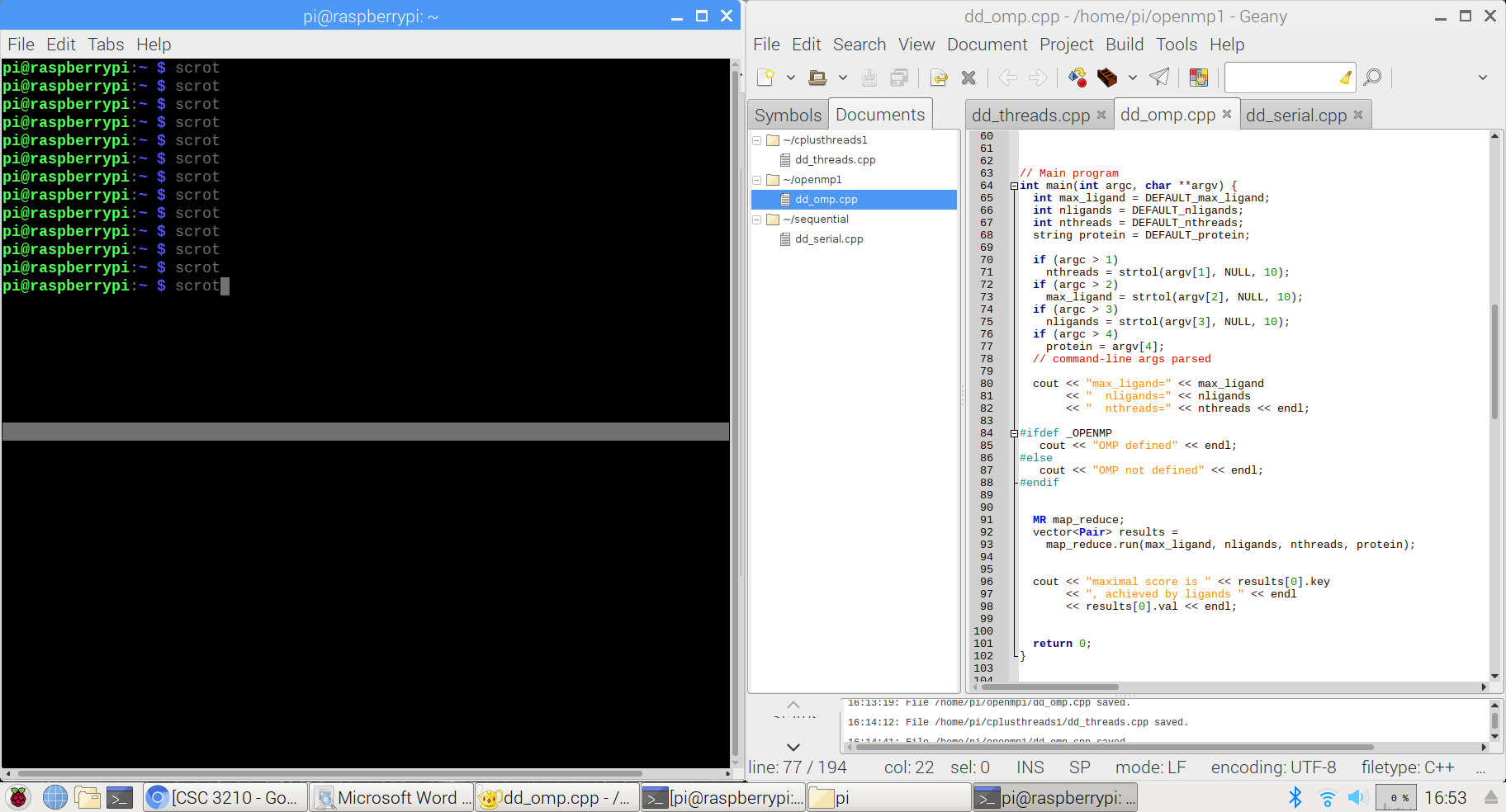


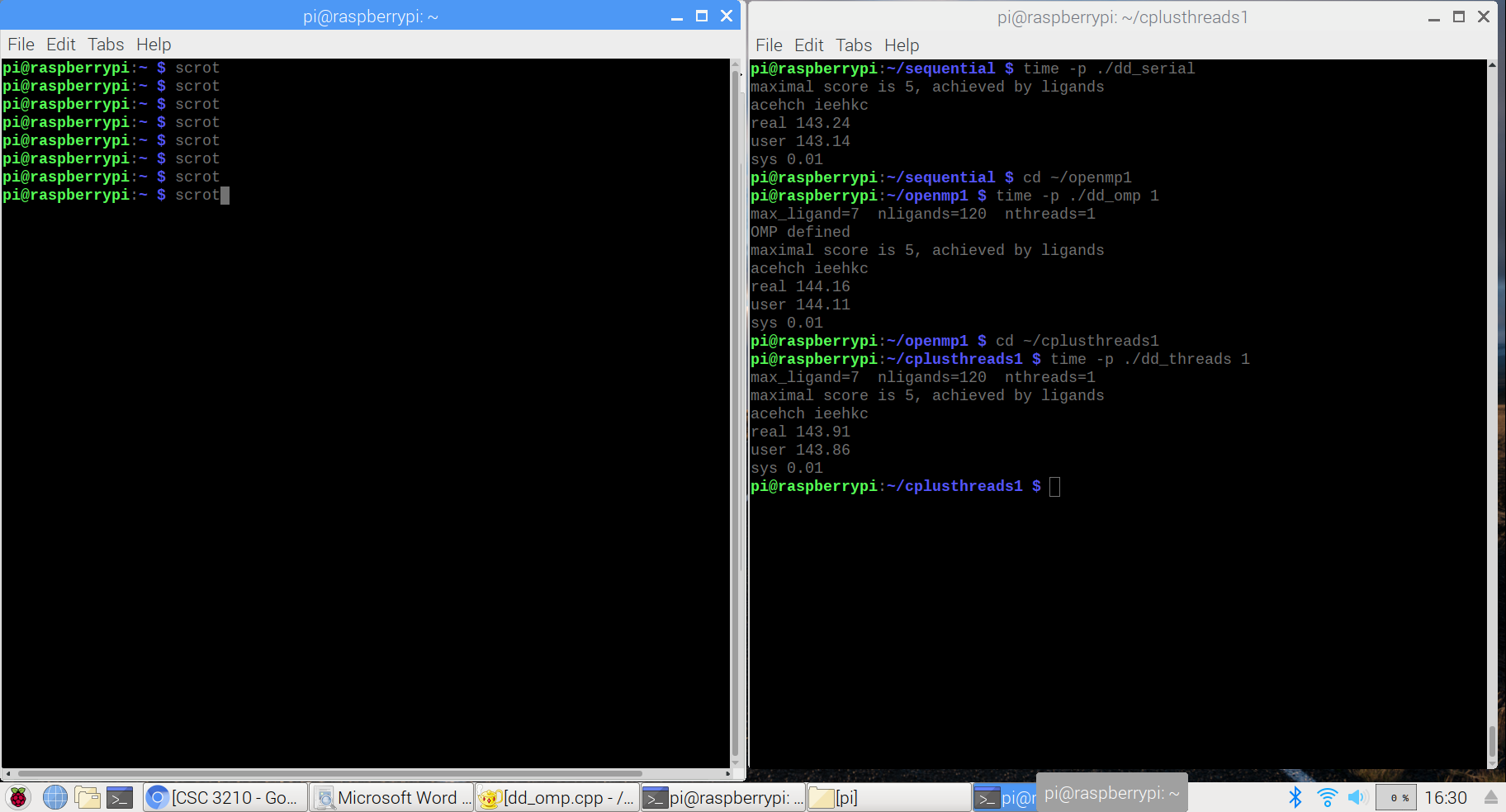
4. C++11 Threads Solution Compilation



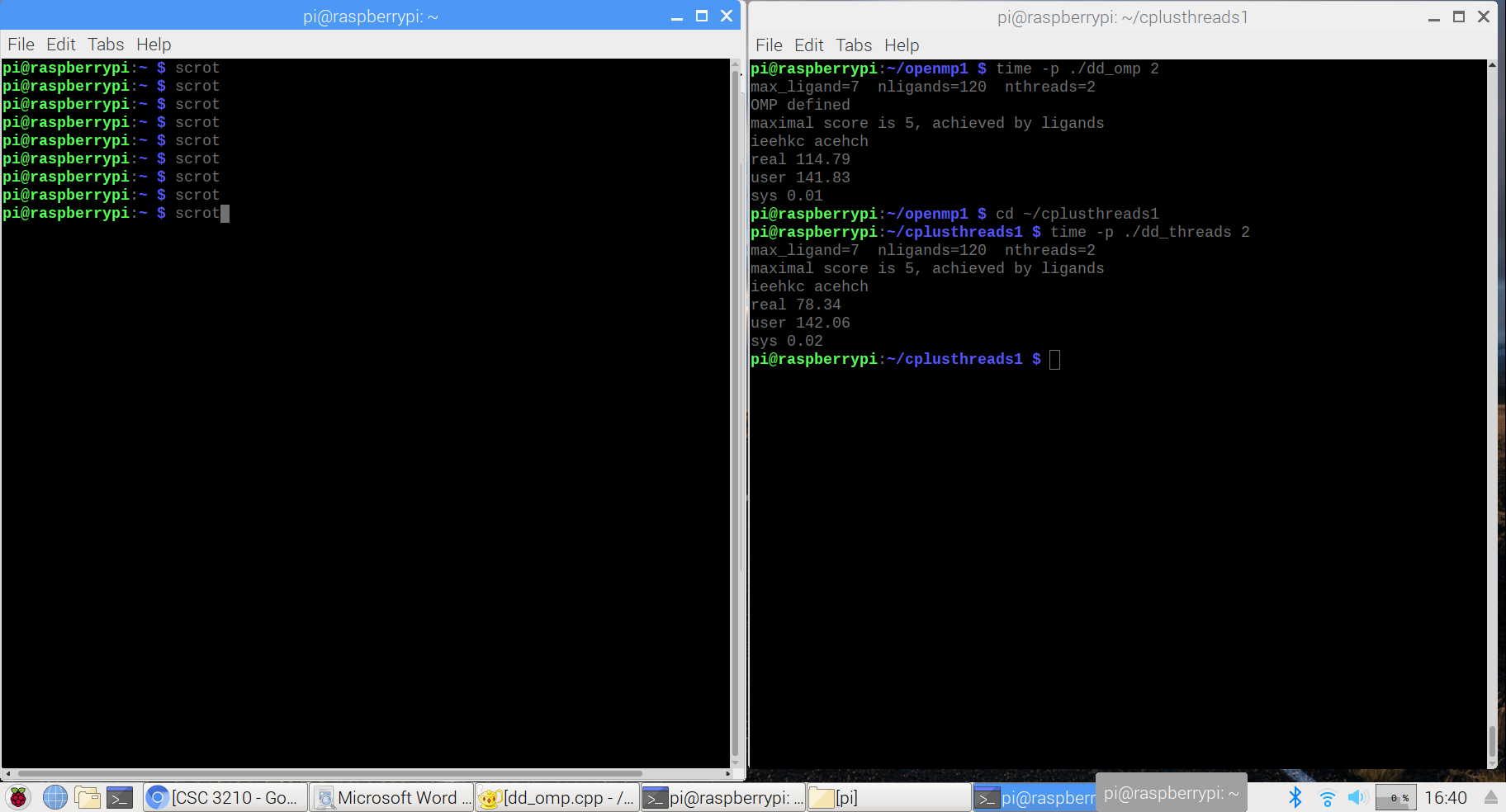
  
5. Measure Run-Time

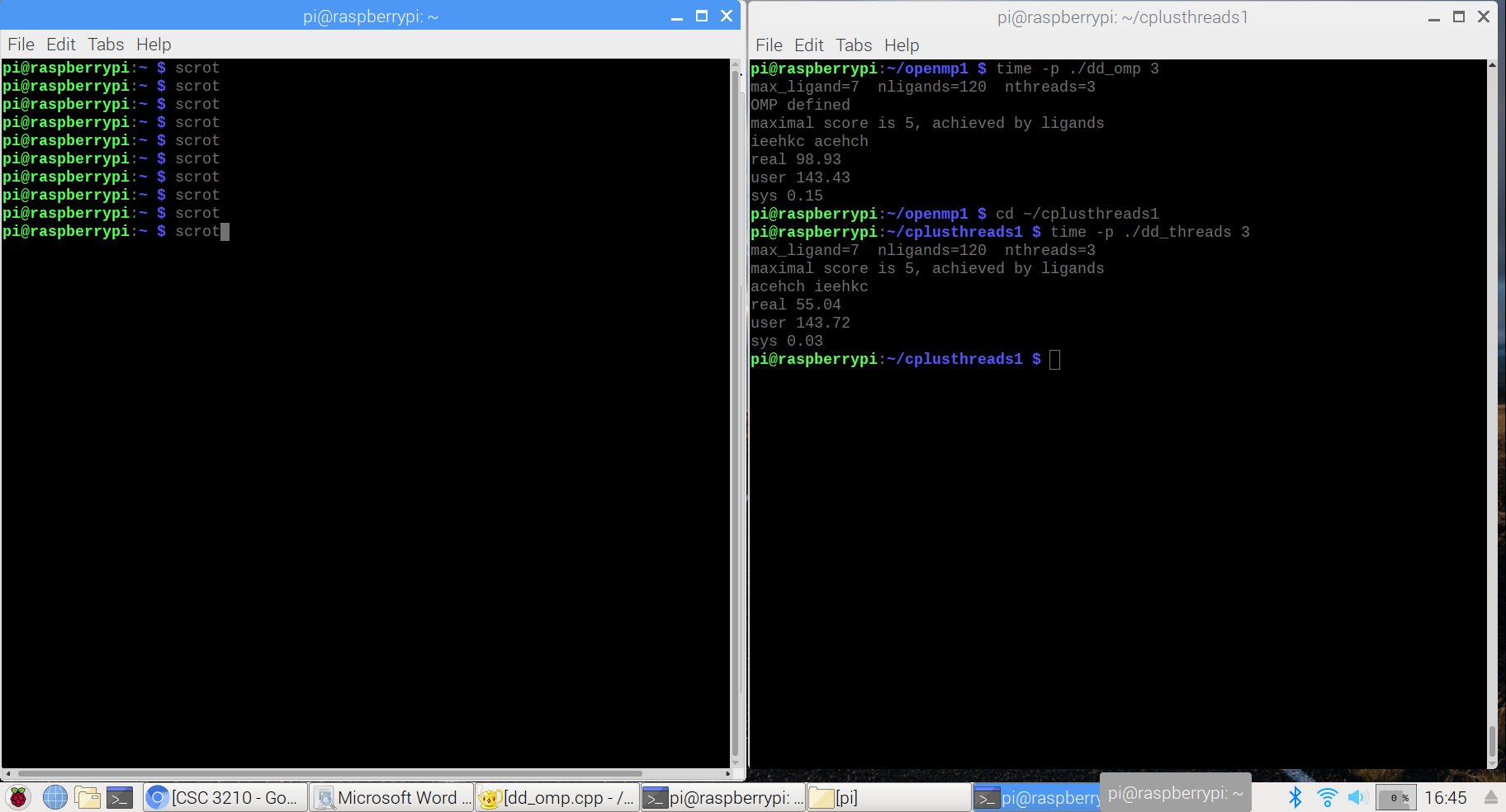


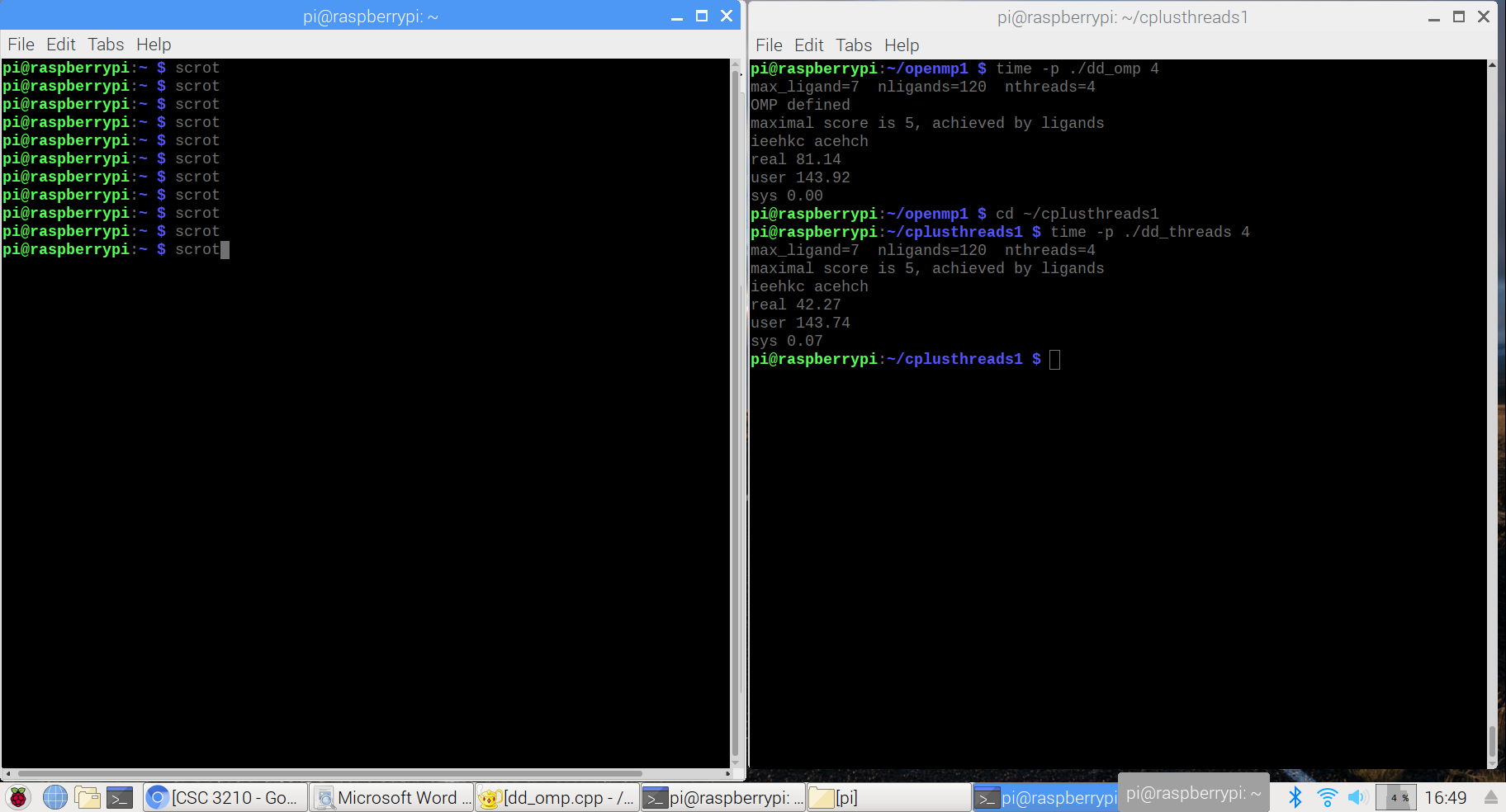
While measuring the running time of each implementation, we have found that the argument of the program is changing the maximum ligand length, and not the number of threads. To make the number of threads as the first argument for both programs, we have edited some codes in cpp files as follows and compile them again.



|  |  |
| --- | --- |
| Implementation | Time (s) |
| dd\_serial | 143.24 |
| dd\_omp | 144.16 |
| dd\_threads | 143.91 |







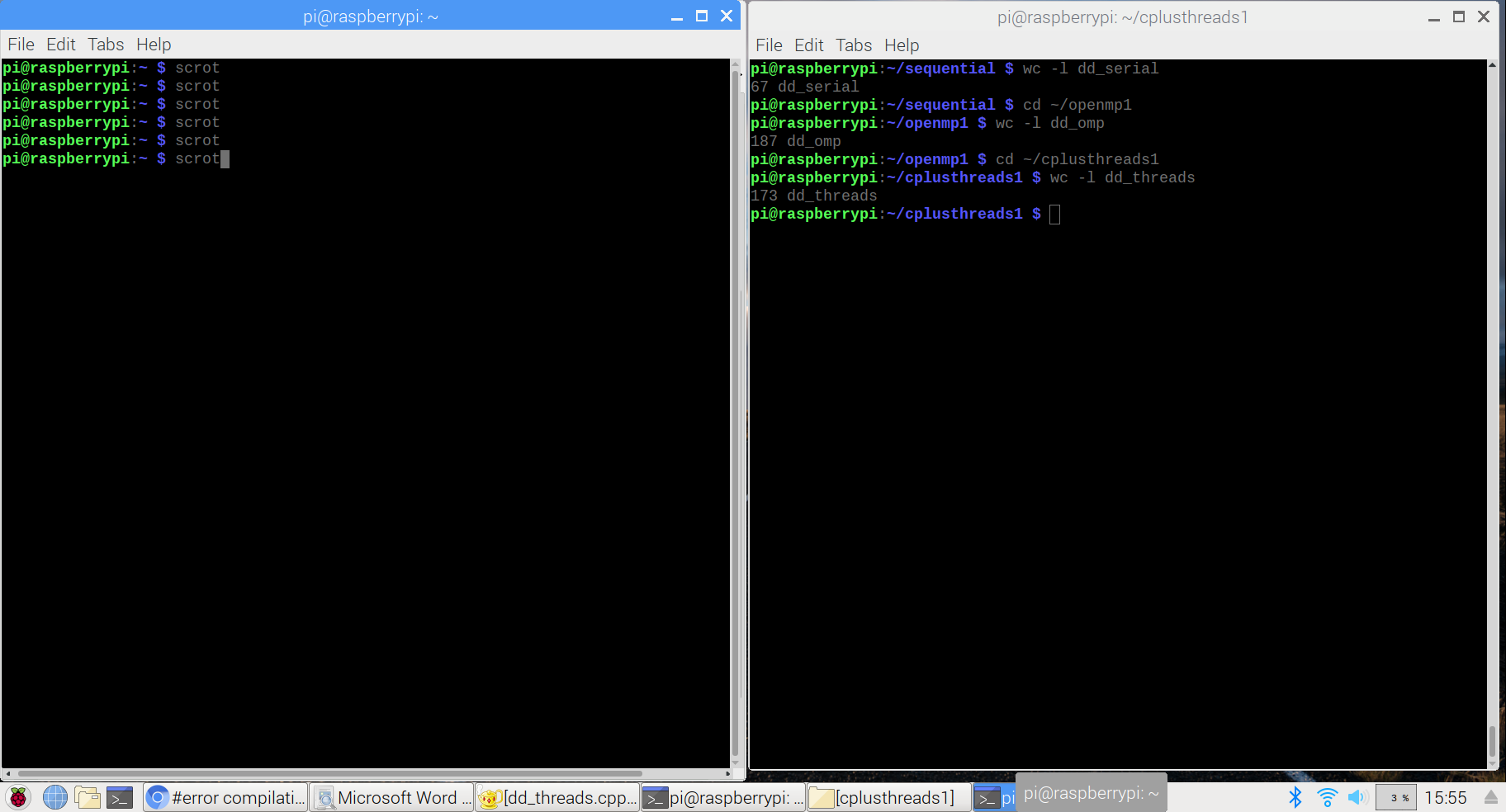
|  |  |  |  |
| --- | --- | --- | --- |
| Implementation | Time (s) 2 Threads | Time (s) 3 Threads | Time (s) 4 Threads |
| dd\_omp | 114.79 | 98.93 | 81.14 |
| dd\_threads | 78.34 | 55.04 | 42.27 |

Discussion Questions

1. Which approach is the fastest?

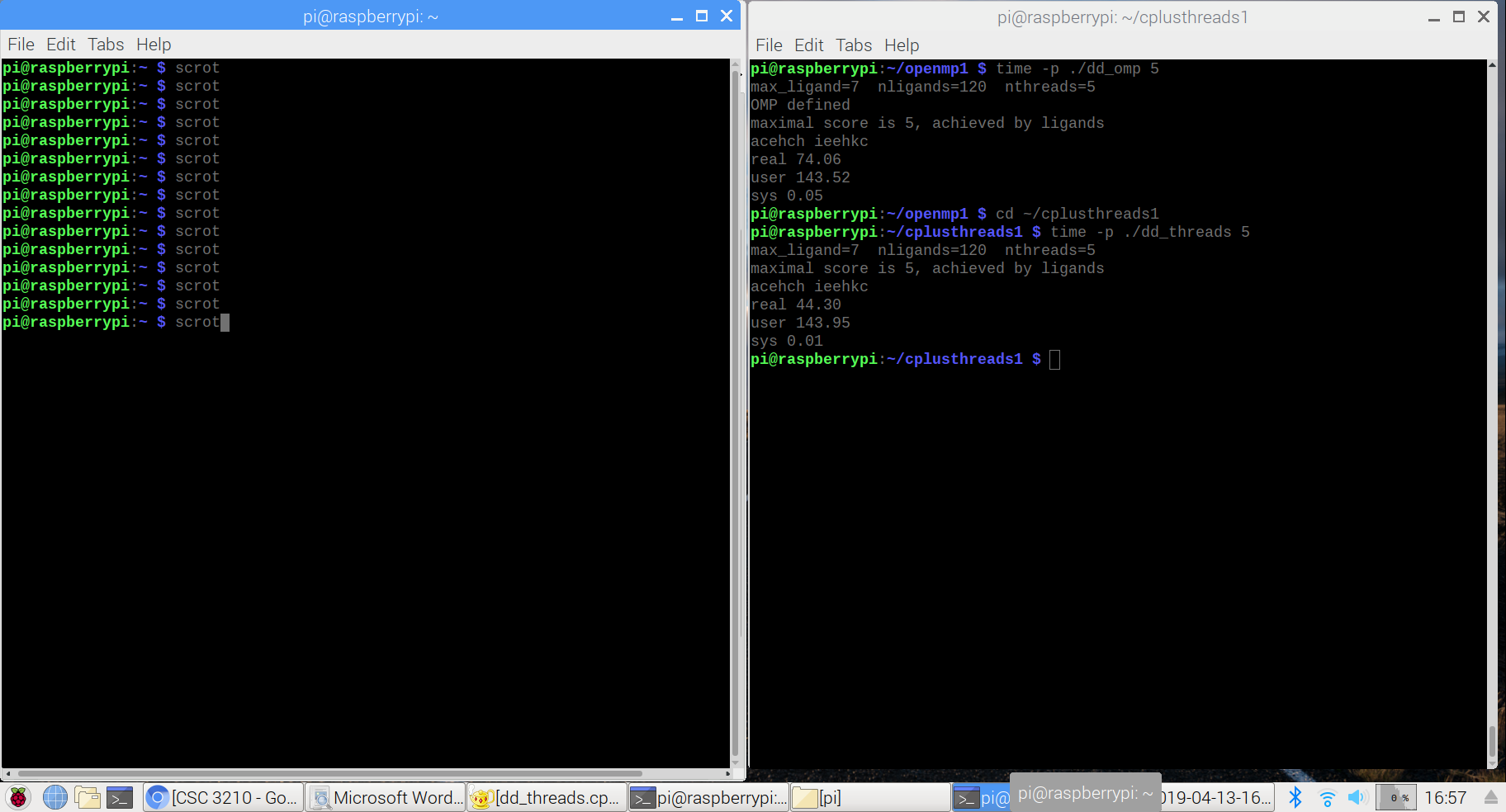
* C++11 threads solution is the fastest approach.

1. Determine the number of lines in each file. How does the C+11 implementation compare to the OpenMP implementations?



* Sequential solution: 67 lines
* OpenMP solution: 187 lines
* C++11 Threads solution: 173 lines
* C++11 solution has fewer lines in the file than OpenMP solution.

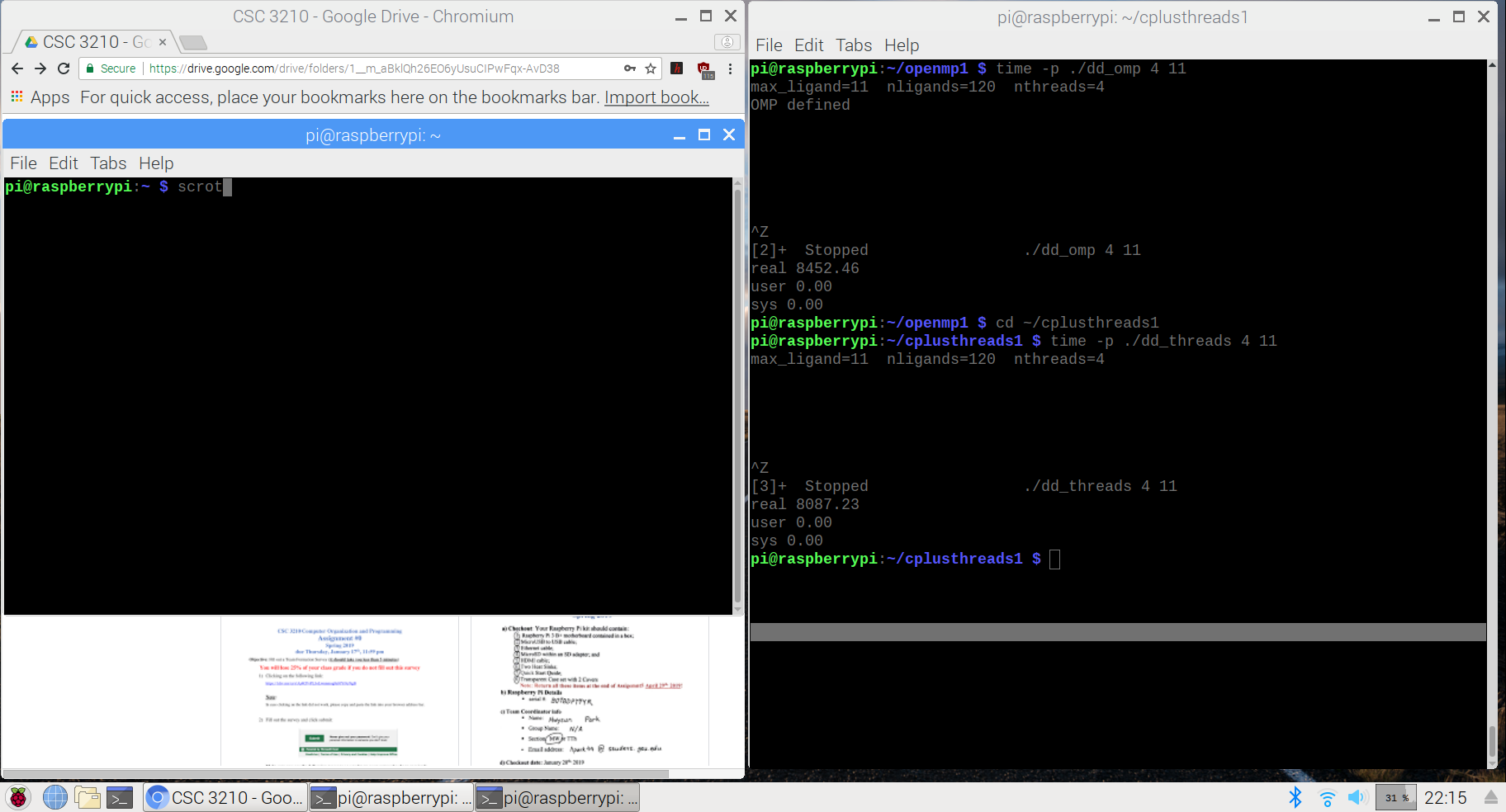
1. Increase the number of threads to 5 threads. What is the run time for each?



|  |  |
| --- | --- |
| Implementation | Time (s) 5 Threads |
| dd\_omp | 74.06 |
| dd\_threads | 44.30 |

* Since the Raspberry Pi have a 4-core processor, the maximum number of threads available is 4. They are similar to the runtimes measured when nthreads was set to 4.

1. Increase the maximum ligand length to 11 and rerun each program. What is the run time for each?



* OpenMP solution could not finish the task up until 8452.46 seconds (2.35 hours).
* C++11 solution could not finish the task up until 8087.23 seconds (2.25 hours).