Hyoungjun Lee  
4/24/20  
Computer Org Programming  
Project A5

Task 3A)

**What are the basic steps (show all steps) in building a parallel program? Show at least one example.**

1. Identify groups of task that can be computed in parallel
2. Check all the task are independent and delegate each partition effectively
3. Managing communication, scheduling, of task to be executed correctly, to utilize parallel programming

**What is MapReduce?**

MapReduce is a program model that allows process large dataset parallel, builds the map and takes sequence of values as input

**What is map and what is reduce?**

Map is a process of producing, reassembled into a new data set made up of smaller data , reduce task, processing the data from map. Make values to from smaller set of values and returning into final data

**Why MapReduce?**

good for writing scientific programs, and processing data more simply adding up an additional computing nodes

**Show an example for MapReduce.**

Could be a drug picture and design in Part B shown below

**Explain in your own words how MapReduce model is executed.**

When we call MapReduce function from library, it will make a large data set into smaller pieces, to be processed over multiple machines, it will create a copy of the same program, the first copy would be master, and rest of copies would be worker, it will reduce stage takes the map data

**List and describe three examples that are expressed as MapReduce computations.**

**Count of URL Access Frequency :** map function will take log of URL access to requests as input, then reduces it by adding up to give the total URL pair

**Inverted Index** : takes a text as input , and it will return a list of <word, page number>,

The output shows how many times each word appears in the text.

**Distributed Grep** : this function takes a string as input, and look up for the pattern inside dataset, and put it together if founded.

**When do we use OpenMP, MPI and, MapReduce (Hadoop), and why?**

OpenMp used for parallel programming shared memory , it has wide usage for all types of coding, allow multiple processor to process small tasks, should be done in sequential.

MPI : stand for Message Passing Interface, routines used for developing parallel code over multiple machines runs over more than one machine

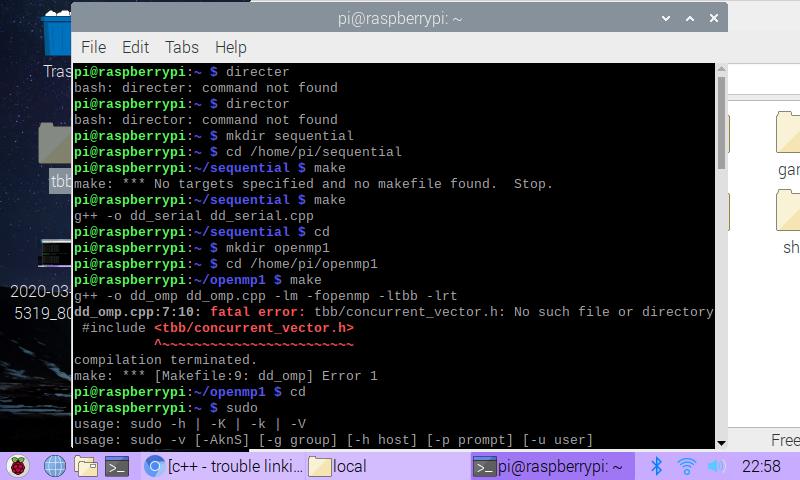
Map Reduce : this function used for for distributed parallel programing making it incredibly reliable MapReduce has a higher fault tolerance than similar systems

**In your own words, explain what a Drug Design and DNA problem is in no more than 150 words.**

This Drug design compared to code with that both can generate ligands for new drugs, testing ligands, small molecules which are capable of binding to proteins . Can be done in parallel to speed up

PartB)

I ran the each implementation first, creating directory for sequential and for OpenMp1, to design drug style programming, but there was some error inside TBB, and stack overflow thinging, I need to install some libraries to figure out error



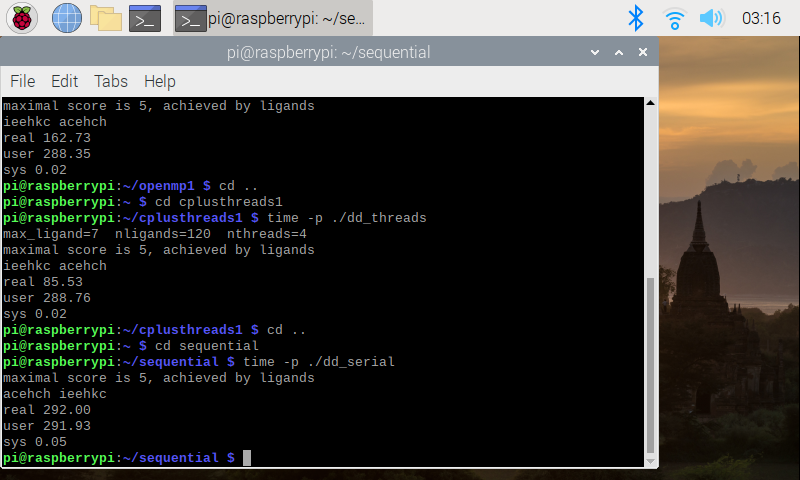
The code I used for is searching the top ligands, installed TBB libraries , and everything works finely

A screenshot of a cell phone

Description automatically generated

As shown in picture, installed it, and compiled

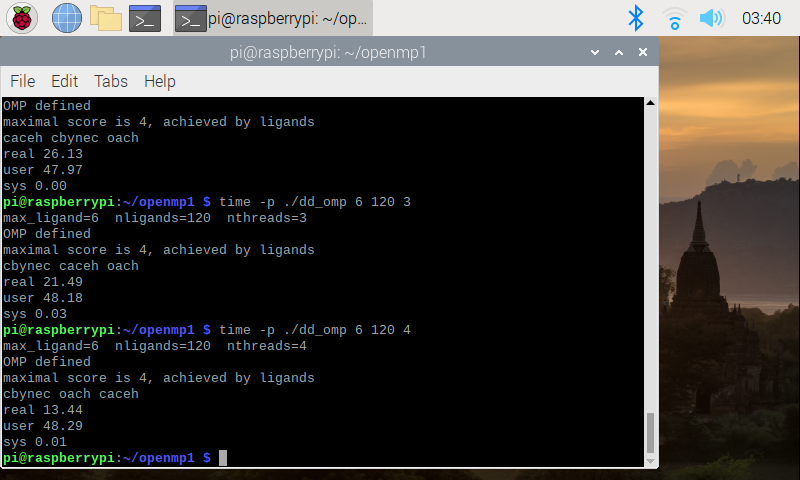
Next, TBB was done in OpenMp make ran quickly, it took 0.02 seconds to finishing sequential



A screenshot of a cell phone

Description automatically generated

|  |  |
| --- | --- |
| Implementation | Time(s) |
| dd\_serial | 292.00 |
| 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 | 0.13 | 0.20 |
| dd\_threads | 0.03 | 0.15 | 0.17 |

Discussion question

1. What approach is the Fastest?

The fastest one is C++11 , when time at 4, beating OpenMp sequential

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

Implementation with 137 to the OpenMP’s 152 lines.

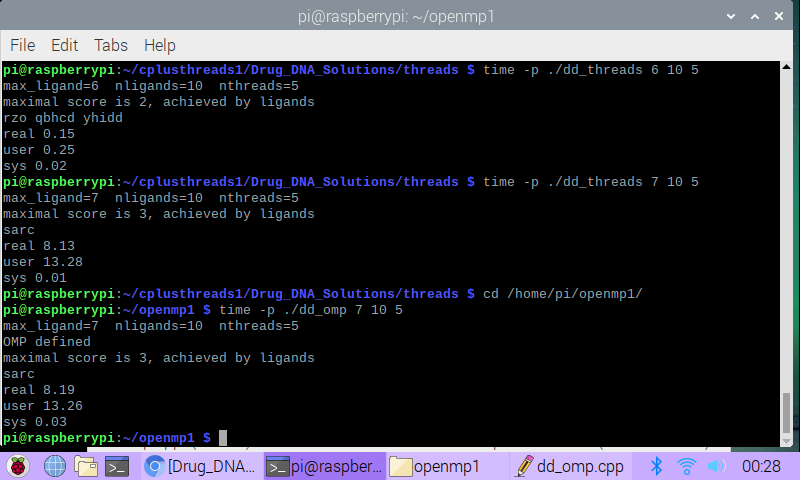
3.Increase the number of threads to 5 threads. What is the run time for each?

A screenshot of a cell phone

Description automatically generated

|  |  |
| --- | --- |
| Implemetation | Time(s) 5 threads |
| dd\_omp | 2.07 |
| dd\_threads | 0.79 |

1. Increase the maximum ligand length to 7 and rerun each program. What is the run time for each?



|  |  |
| --- | --- |
| Implemetation | Time(s) |
| dd\_omp | 8.19 |
| dd\_threads | 8.13 |