Homework 03

Fall 2019

CISC 372

Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Total 4 questions. 25 points per question.

1. Upload your Jupyter Notebook for HelloWorld code on Canvas. Make sure to save your file as <firstname>.ipynb (ipynb being the extension for Jupyter Notebook). When the TA/myself grade your assignment, we should be able to open the notebook (8 points), build (8 points), run and see a correct output (9 points).

The output being “Printing helloworld from ThreadXX” and this should appear X number of times depending on how many threads you have allocated.

Use EECIS system as your baseline machine. You are welcome to use your laptop if you have OpenMP installed. If you did so, please mention your laptop configuration in the assignment. Most important how many cores does your laptop have. (use the command line options you must have figured out from homework assignment 2)

(25 points)

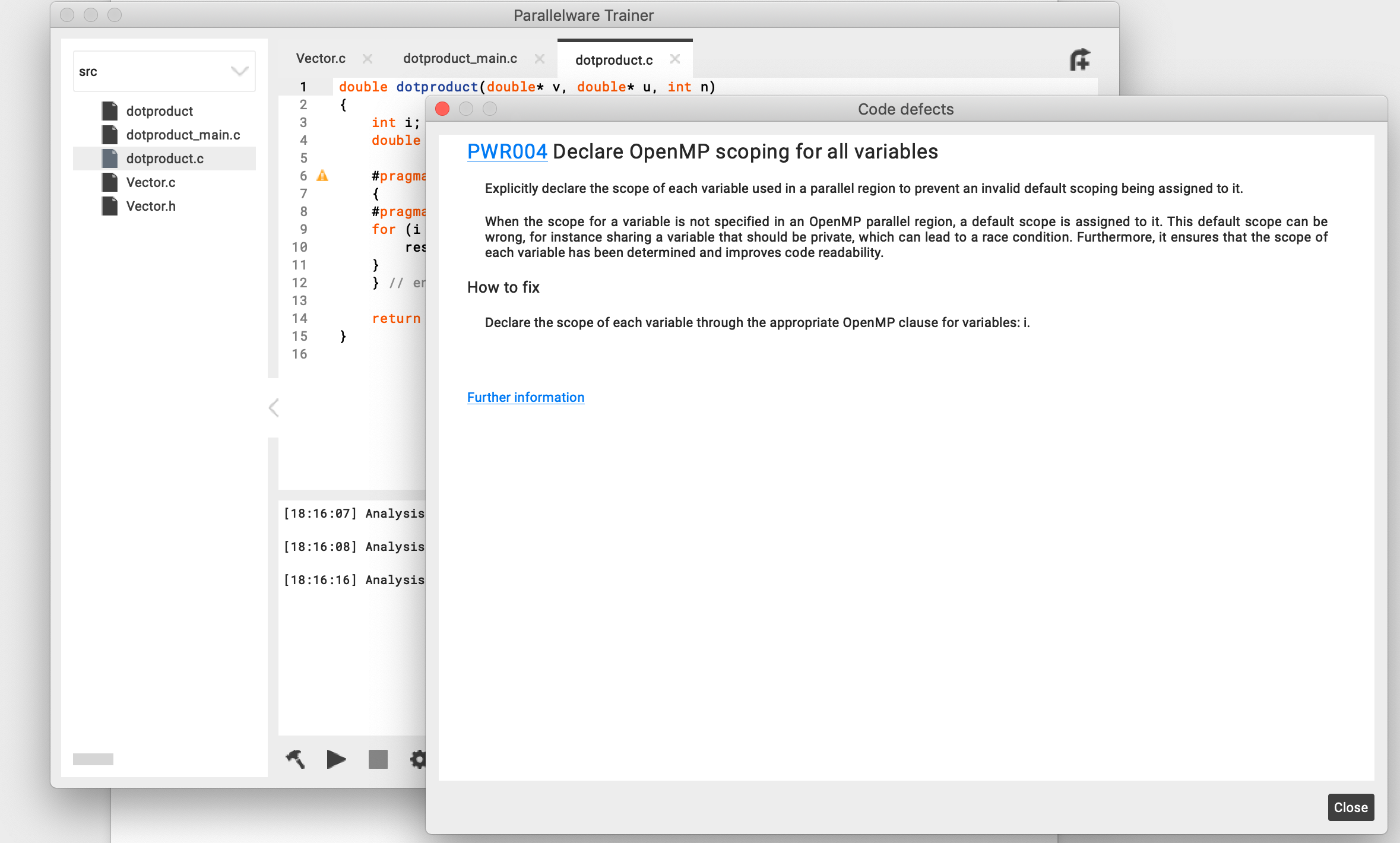
1. Use the Appentra tool either on your laptop (if you can use OpenMP compiler on your laptop) or the tool on the EECIS machine and learn how pi.c are parallelized. Attach screenshots of your output. 25 points (24 points + 1 point for neat presentation of your results)
   1. For pi.c run the code for 2 diff. inputs. For example ./pi 10000 and ./pi 50000
   2. Use word document to document your output and screenshots.
   3. Your answers should include output for 1, 2, 4, 8 threads
   4. Give you answer in the form of a table. Like below
   5. If you do this right, you will earn 3 points per row

| Code | Number of threads | Execution Time taken |
| --- | --- | --- |
| ./pi 10000 | 1 |  |
| ./pi 10000 | 2 |  |
| ./pi 10000 | 4 |  |
| ./pi 10000 | 8 |  |
| ./pi 50000 | 1 |  |
| ./pi 50000 | 2 |  |
| ./pi 50000 | 4 |  |
| ./pi 50000 | 8 |  |

1. Use the Appentra tool either on your laptop (if you can use OpenMP compiler on your laptop) or the tool on the EECIS machine and learn how dotproduct.c are parallelized. Attach screenshots of your output. 25 points (24 points + 1 point for neat presentation of your results)
   1. For dotproduct.c run the code for 2 diff. inputs. For example ./dotproduct.c 50000 ./dotproduct.c 15000
   2. Use word document to document your output and screenshots.
   3. Your answers should include output for 1, 2, 4, 8 threads
   4. Give you answer in the form of a table. Like below
   5. If you do this right, you will earn 3 points per row

| Dotproduct | Number of threads | Execution Time taken |
| --- | --- | --- |
| ./dotproduct 500000 | 1 |  |
| ./dotproduct 500000 | 2 |  |
| ./dotproduct 500000 | 4 |  |
| ./dotproduct 500000 | 8 |  |
| ./dotproduct 15000 | 1 |  |
| ./dotproduct 15000 | 2 |  |
| ./dotproduct 15000 | 4 |  |
| ./dotproduct 15000 | 8 |  |

Note: The Appentra tool gives full compilation output and explanations to how certain things are done. For example see the following screenshot on “code defects” which is the output of the little triangle in the background image. So make sure to press thru the buttons and the compilation outputs to learn the best you can using this tool. Questions from both the codes on how OpenMP was used to parallelize may be asked in the midterm.



1. Fill up this google form on what you liked and did not like about the tool.

3 bullets each for pros and cons = total 6 bullets.

4 points per bullet = 24 points + 1 free point for giving feedback!

<https://docs.google.com/forms/d/1OznYTYC5vKp2IQBhgSGNiAdSEFDE85GRYPoGFXN4xq4/edit>