Engineering Parallel Software Self-Appraisal Form
Also known as the ``I am great!''

INSTRUCTIONS

This document is the self-assessment of an individual's performance. This, together with your project leader's final assessment, form the appraisal package. Here you will describe your own activities, strengths, and weaknesses and give your own numerical assessment of your performance and the piece of the team pie you deserve.

The numerical scale for performance is:

5   Extraordinary contributor: performed exceptionally and had a very high-impact on project. A ``project making'' team member.

4   Exceeds expectations: performed well above expectations on all portions of project.

3   Meets expectations: Solid contributor that performed adequately on all aspects of the project.

2   Improvement needed: Performance on the project was lacking in important regards. Underperformance by the member reflected in weaknesses in the project that either had to be compensated for by other members, or reflected in the final project result.

1.   Unacceptable: Underperformance in the project significantly increased the burden of other members.

Signed, self-appraisals are due in the assignments box at Piazza and to your project leader by 5PM 12/2/2016.

TEAM MEMBER NAME Vy-An Phan	TEAM/PROJECT NAME: NMS NUMBER OF TEAM MEMBERS: 6	ELEMENT OF PROJECT OMP/SIMD
PROGRAMMING BUDDY(S) Tushar, Simon, Jerry, Seth	Overall Performance score (5-1): 4	Participation Percentage: 20%

1) GOALS AND RESULTS: Evaluate performance on the project (60% of total score); Score (1-5): 4 Document any discrepancies below:

Portions of project you were responsible for and you delivered on:
    Translated base code from Python to C
    Helped set up nmsModule skeleton
    Wrote OMP and SIMD implementations
    Wrote/formatted tests.py, cleaned up/formatted code
    Helped debug GPU + ran tests
    Set up/wrote powerpoint presentation + drew the graphics

Portions of project that you were responsible for, but failed to deliver:
    I did everything that I needed to do.
    I did attempt to do a few extra things that we ultimately decided we didn't need, like a C translation of the C++ module and a size-4 SIMD vector implementation (we ended up using the size-8 one).

2) TEAM WORK/TEAM LEADERSHIP (20% of total score): Assess your own ability work with team members – as a team member did you show: reliability, flexibility, accountability, friendliness? As a project leader (if applicable) did you show: honesty, competence, leadership? Finally, comment on your own ability to communicate clearly (or not). **Give overall score (5-1) here: 5**
Reliable: I showed up to all the required project check-ins, and independent group meeting times. Flexibility: I was willing to stay behind late on some days to get extra things done. Accountability: I did everything that I had to do, except for a few extra experiments which we didn't use anyway. Friendliness: As a whole, our team didn't have any problems/arguments. I don't think I had any problems communicating. I made sure to tell my teammates exactly what I was doing, and gave suggestions and help when it was asked for.

3) (20% of total score) AREA OF STRENGTH/: Assess your own areas of strength.
AREAS FOR IMPROVEMENT: Describe your areas of weakness.
The purpose of this latter area is less evident in a semester project. In an on-going working relationship these areas help the employee and project leader/manager utilize strengths and remediate weaknesses. Give score ((5-1) here: 4

SOFTWARE SKILL AREAS	**Particular areas of strength (choose from categories on far left)**	**Particular areas for improvement (choose from categories on left. Give a one sentence plan for improving any weaknesses.)**

1. Correct coding
   a. Understanding invariants
   b. Test before you code
   c. Applying unit testing, integration testing, system testing
   d. Refactoring
   e. Daily system and Regression testing
2. Computationally efficient coding
   a. Efficient coding techniques
   b. Good algorithms
   c. Problem formulation
   d. Profiling of code
   e. Refactoring
   f. Use of parallelism
3. High-productivity coding
   a. Reuse of software
   b. Do the simplest thing that could possibly work
   c. Ability to quickly use program generators, third party software, or other programming tools
4. Facility with Patterns
   a. Overall architecting ability
   b. 'Structural and computational patterns
   c. Parallel Algorithm Patterns
   d. Implementation Patterns
   e. Target HW Patterns
	■ **1b, 1c, 1d, 1e.** I did a LOT of testing when doing this project, especially for the OMP-SIMD stuff.
	
	■ **2a-f.** I did the usual stuff, like code reuse, keeping my code organized, making all updates in proper files for modularity. We did a lot of work in thinking about which algorithms would work best, and how much parallelism vs. serial we wanted. I also helped write the test file which printed out our end results and speedups.
		■ **3b.** I did a lot of exploring in this project, including trying out some new things that I knew in the back of my head probably weren't going to work – next time, I'll stick to improving the best of the logical attempts we already had.

4) EVALUATED *MEMBER*/(EMPLOYEE) DISSENTING COMMENTS: This is an area for the person being evaluated (not the evaluator) to write dissenting comments regarding this review. If there are any ways in which you feel the review is inaccurate write them here. (Attach an additional document as needed. Be sure to note below if there is an additional document.)

SIGNATURES   <u>**Signature**</u>   <u>**Date**</u>

Project member   29 Nov. 2016
Signature indicates above discussed with project leader (or evaluating team member during project leader review).

Project Leader/Evaluating Team Member _____   29 Nov. 2016