COMP 533 Final

Fri, May 5, 2017, 12pm-3pm

Instructions
1. Please spread out and try and sit in alternate seats.
2. This is a closed book exam.
3. Write on the exam itself.
4. Write on the opposite side of a question page if there is not enough space to solve a problem.
5. There are:
   - 6 numbered pages including this one; there are no blank numbered pages.
   - 2 questions.
   - 150 possible points. Point values appear next to each question.
6. You have 3 hours.
7. Explain all answers.
8. If you need to make any assumptions to clarify a problem, write your assumptions down. Only reasonable assumptions get full credit.

Name (in Capitals) and Onyen (in Capitals)
____________________________________

Pledge: I have neither given nor received unauthorized aid on this exam.
(signed)____________________________________

Please do not write below
1. _____/90	2. _____/60

Total: _____/ 150
1. **Layering and RPC Implementation**
   a) [15 pts.] If a set of modules, A, must be layered over another set of modules, B, then what constraints must be satisfied by the code in A and B? Be as specific as you can and define all terms you use in your answer.

b) [15 pts.] What are the layers in the implementation of GIPC RPC, that is, give names to these layers and describe what functions they provide. You do not have to describe how a higher-level layer uses the facilities of lower-level layers – simply identify the functions added by each layer, starting from the bottom.
(c) [60 pts] Give the advantages of layering in general and layering in GIPC RPC in particular. That is, give each abstract advantage of layering and illustrate it using GIPC layering. Be as specific as you can. The more examples you can give in the illustration, the better. If possible, draw examples from your course experience (lectures and assignments).
2. [60 pts.] **Consensus**

Distinguish each of following consensus algorithms (discussed in class and/or assignment 6 and implemented in GIPC) by giving some combination of features that make it unique. In other words, give some combination of features in each algorithm that does not occur in other algorithms mentioned below. You can assume, and thus do not have to define, the definition of proposers, acceptors, and learners. Also only give those aspects of the learn, accept, and prepare phases that are needed to make a distinction among the algorithms. Mention features that have to do with the steps in the algorithm rather than the guarantees given by the algorithms. Thus, you do not have to explain or address what the guarantees of atomicity or synchronicity mean. The distinguishing features can describe both what the algorithm does and does not do.

a) **Asynchronous Non-Atomic**

b) **Synchronous Non-Atomic**

c) **Asynchronous Atomic**
d) Synchronous Atomic

e) Basic Paxos

f) Sequential Paxos