CS110 - Principles of Computer Systems

Midterm Exam

(Total time = 50 minutes, Total Points = 50)

Name: (please print)_____

In recognition of and in the spirit of the Stanford University Honor Code, I certify that I will neither give nor receive unpermitted aid on this exam.

Signature:_____

This examination is close book and close notes. You may not collaborate in any manner on this exam. You have 50 minutes to complete the exam. Please write your answers on the exam. Note there is one problem per page so the amount of space provided does not necessarily provide an indication of the expected length of the answer. In other words, do not feel compelled to fill every nanoacre of the exam with writing. Before starting, please check to make sure that you have all 7 pages.

Question	Points	Score
1	6	
2	7	
3	8	
4	7	
5	7	
6	9	
7	6	
Total	50	

1. (6 points) The course reader describes four generic categories of problems faced by systems. The categories were: *Emergent properties*, *Propagation of effects*, *Incommensurate scaling*, and *Trade-offs*. Which of these categories is helped by using a client and service architecture for a system? Justify your answer.

2. (7 points) While running the Windows operating system on a simulator it was noticed that there were many procedure calls to routines that did nothing else but call another routine. The source code for these "null" procedures must of looked something like:

```
returnType
RoutineA(arglist)
{
    return RoutineB(arglist);
}
```

where *arglist* is the list of arguments to the function and *returnType* is the type of the return value.

The chain of "null" procedures sometimes reached up to dozens of calls before reaching a routine that actually did some real work. This seemed strange since it would be equivalent functionality to have the original routine directly call the routine that performed the work. It was further noted that the routines themselves were widely spread in the kernel like they were each coming from a different module of the kernel. This structure is related to one of the following concepts from the course. Identify the concept and explain why this is the case.

a) Naming conflict
b) Layer bypass
c) Enforced modularity
d) Deadlock avoidance

Be sure to justify your answer.

3. (8 points) For each of the following elements of a computer system, state if the object would be best classified as a *memory*, *interpreter*, or a *communication link*. Briefly (a single sentence) justify your answer.
    a. Hash table
    b. Java Virtual Machine
    c. File
    d. x86 processor
    e. Firefox browser
    f. gcc compiler
    g. Microsoft Word editor
    h. The Internet

4. (7 points) My cell phone behaves differently depending on my location. When I'm at Stanford I can call my office by dialing 7230474 but when travelling on the USA east coast I have to dial 16507230474 and while in France I had to dial +16507230474. Which of the following terms used in our discussion of naming best describe what is happening here? Be sure to explain your answer.
    a. Name space
    b. Name mapping algorithm
    c. Universe of values
    d. Naming context

5. (7 points) What are the tradeoffs between using a RPC versus a procedure call for accessing a module? Your answer should include at least one advantage and one disadvantage for each of the approaches.

6. (9 points) For each of the C language statements below, state if the statement has a race condition if multiple threads execute it simultaneously. You can assume that X is an C integer data type (e.g. char, short, int, long, or long long). Function getint() and printf() are assume to be thread-safe (e.g. multiple threads can call into it without race conditions).
    a. X = X + 1;
    b. X = getint();
    c. { int tmp; tmp = X; printf("%d\n",tmp); }

7. (6 points) A famous database researcher criticized the Unix file system for what he called the “double fault” problem. His database system kept its data in a very large file in a Unix file system. The researcher noticed that accesses to the database file seemed to generate twice the number of disk read requests as compared to the number of file read system calls going to the database file. This problem did not occur if the database did not go through the file system and instead stored data directly on the disk. Using your knowledge of the Unix file system implementation explain how this could happen.