CS35L – Software Construction Lab
Winter 2016
Final Exam

Date: 16th March 2016
Time: 11.30AM to 2.30PM
Total points – 100
Duration – 3 hours

Student Name: _____ UID: _____

Instructions:

I) Please write your response clearly in the allocated space. Unreadable answers will NOT be graded. It may help to first formulate and write your answers on the back of each sheet. The space provided on the front of each sheet, should have just the answer.

II) This examination is open-book, open-notes. Any printed or hand-written materials are permitted. **Electronic devices are NOT permitted.**

III) If you have questions, raise your hand and TA will come to you and clarify the question for you. Any questions other than clarification of the exam will not be answered.

1) [12 points] Regular Expression - In each of the following cases, you are given 2 regular expressions R1 and R2. You have to say if they generate the same set of strings or not. If they generate the same set of strings then say YES **and** provide an example of a string generated by them. Otherwise say NO **and** provide an example of a string which is generated by one, but not by the other expression. (Parentheses '()' are used for grouping regular expressions.)

- R1 is **(xy|x)+x** R2 is **x(y|x)+x**

- R1 is **[ab]*** R2 is **(a*b*)***

- R1 is **(a|ba)?** R2 is **(b)?a?**

2) [4 points] - Mention any 2 major differences between static linking and dynamic linking.

3) [8 points] Briefly explain how buffered I/O works at a conceptual level using the example of getchar(). How is it able to improve program performance?

4) [6points] What are thread-safe functions? Explain by implementing a very simple thread-safe function.

5) [4 points] – How does locale affect the behavior of 'sort' command? Explain with sorting examples.

6) [20 points] Multithreaded Sort – You are required to sort a set of 100 million integers in descending order. This can take a very long time if your program is single threaded. Your task is to write a multi-threaded program, using pthreads, that distributes the workload across a set of 10 threads. Some starter code is given below. The standard library **qsort**() function, with a compare() function, should be used to do the sorting per thread. You don't have to rewrite anything that is already given below.

```
#include<stdio.h>
#include<math.h>
enum { RECORD_COUNT = 100000000 };
enum { NTHREADS = 10 };
//Your code here
int main(int argc, char** argv) {
      int *record = (int*) malloc(sizeof(int) * RECORD_COUNT);
      if(!s) {
            fprintf(stderr, "Cannot allocate memory!\n");
            exit(-1);
      }
      // Assume the following function call loads values into 'record'
      ReadRecords(record);
      //Your code here
      // Assume the following function call prints the sorted values
      PrintRecords(record);
      free(s);
      return 0;
}
```

7) [9 points] Multi-threading - Here are the outputs of the 'time' command when a multi-threaded version of a ray-tracer program was executed on the 16 core SEASNET server.

Threads used : 1
real 0m31.698s
user 0m31.607s
sys 0m0.018s

Threads used : 2
real 0m16.403s
user 0m31.796s
sys 0m0.017s

Threads used : 4
real 0m10.803s
user 0m31.750s
sys 0m0.017s

Threads used: 8
real 0m7.012s
user 0m32.989s
sys 0m0.015s

- Why is the 'user' time greater than the 'real' time in some of the cases?

- What could be a possible reason for the 'user' time being almost the same in all the cases?

- What could happen to the above times, if the Seasnet administrators set a policy that allowed your programs to use atmost 2 of the cores?

8) [10 points] You are required to allocate/deallocate memory to a 3-dimensional array using malloc() and free() functions. You are given a “pointer-to a-pointer-to a-pointer” as shown below. The sizes for the 3 dimensions are 100, 200 & 300 respectively. Write 'C' code that allocates memory using malloc(), followed by code to deallocate the same memory using free().

float ***M;

9) [2 points] GIT – Describe any one major difference between doing a 'git rebase' versus doing a 'git merge' on a repository.

10) [10 points] SSH -When you logged onto your partner's machine in a “password-less” manner, describe (**conceptually**) how the server validated the client (you) using encryption schemes we learnt in class. You should list the steps involved.

11) [6 points] GIT – If you are the Project Manager for a group of software engineers, then why would you want your team to maintain all their code on a source control system (GIT, SVN, etc)? Briefly describe 3 different reasons.

12) [6 points] GIT - You are working on code that is being tracked by GIT on your local machine. You just made some modifications to files (bar.c and bar.h), and also added a new header and a new source file (foo.h and foo.c). You haven't committed these changes yet.

- How would you know the differences between the working copy of bar.c and bar.h, and their respective versions in a commit that was made just before the last commit (penultimate commit)? You cannot use commit hashes to refer to commits.

- How would you commit all the changes that you just made to your code (files bar.c bar.h foo.c foo.h)?

13) [3 points] You have just built an executable (ELF format) out of a 'C' program that uses shared libraries. The executable runs without any errors while using functions from the shared libraries. But when you run the '**ldd**' tool on the executable, it does NOT show the .so file for one of the shared libs, that contains functions you are using in the program. What is the reason for this? Briefly explain this behavior at a conceptual level. (NOTE:This is not because of the ldd file not being able to locate the .so file in the file system)