CSCI 1312 (Introduction to Programming for Engineering), Fall 2015

Review for FInal Exam

1 Format of the exam

The exam will be at the scheduled time for the course final, December 11 at noon. The exam will be about twice the length of the first exam (hence intended to take about two hours), but you can use the full three-hour period if you like. Like the quizzes, it is “open book/notes”, which means you can consult paper or electronic copies of the textbook and your notes, sample solutions *from this year only*, your own graded work, and anything on the course Web site. You may not use other books, materials from this course from previous years, or a calculator or computer (except as noted below), or (of course) each other’s papers. You may use a calculator or computer to

- Consult allowed sources (e.g., browse class notes).
- Write, compile, and run C programs — if you think this will help you answer questions such as “write some code to ...”. Notice however that you can also answer such questions just by writing out answers, as you’ve done on previous exams/quizzes.

Questions will mostly be similar in format to the ones in quizzes and minute essays — some short-answer or multiple choice, some questions about programs in the exam (in the quizzes, “what does this program do/print”, but since you have access to a compiler this time, they’ll focus on why the code does what it does, and if it doesn’t work how to fix it), some “write a program to do this task” — but probably somewhat longer and/or more difficult.

2 Lecture topics to review

You are responsible for all material presented during lecture, *including the sample programs*, but the following is a list of topics I consider most important.

- Basics of how computers/programs work — source code, object code, executables; text editors and compilers.
- Number systems — converting decimal to binary/octal/hexadecimal and vice versa, including fractions.
- Data representation — basic idea of how negative integers and floating point numbers are represented, and how they differ from the numbers of pure mathematics.
- Variables in C (types, declarations).
- Expressions and statements in C.
- Conditional execution in C.
- Functions in C — defining them, using them, using library functions.
- Loops in C.

- Arrays in C (1D only).
- Sorting and searching, at the level presented in class, and "order of magnitude" of functions.
- I/O in C — using library functions to read from and write to both stdin/stdout and files.
- Pointers in C.
- Strings in C.

Material discussed in class after pointers — dynamic memory allocation, user-defined types, and linked data structures — will *not* be included in the exam.

3 Reading to review

You should have read, or at least skimmed, all of the assigned reading, but the focus will be on material presented or at least mentioned in class.