

CSCI 1320 (Principles of Algorithm Design I), Fall 2008

Homework X

Assigned: December 11, 2008.

Due: December 17, 2008, at 5pm. *Not accepted late.*

Credit: Up to 30 extra-credit points.

1 General Instructions

Below is a list of programming problems. You can attempt as many or as few as you like, but notice that you can receive at most 30 extra-credit points. You will probably learn more by doing problems from later chapters, but if your goal is to maximize points — do whatever seems easiest to you.

I am also open to the possibility of giving extra credit for other work — other problems from the textbook, a report on something course-related (perhaps something related to the history of programming, or other programming languages), etc. If you have an idea for such a project, let's negotiate (by e-mail or in person).

For this assignment, please work individually, without discussing the problems with other students. If you want to discuss problems with someone, talk to me.

Submit your program source (and any other needed files) in the usual way, by sending mail to `bmassing@cs.trinity.edu`, with each file as an attachment. Please use a subject line that mentions the course number and the assignment (e.g., “csci 1320 homework X”). You can develop your programs on any system that provides the needed functionality, but I will test them on one of the department's Linux machines, so you should probably make sure they work in that environment before turning them in.

1. (Up to 10 extra-credit points.) Do problem 49 *or* problem 50 from chapter 6 of the textbook (p. 385).
2. (Up to 10 extra-credit points.) Do problem 51 from chapter 6 of the textbook (p. 385).
3. (Up to 10 extra-credit points.) Do problem 27 from chapter 7 of the textbook (p. 454).
4. (Up to 10 extra-credit points.) Do problem 29 from chapter 8 of the textbook (p. 545).
5. (Up to 10 extra-credit points.) Do problem 34 from chapter 8 of the textbook (p. 548).
6. (Up to 10 extra-credit points.) Do problem 34 from chapter 8 of the textbook (p. 548), but using binary search *written using recursion rather than a loop*.
7. (Up to 10 extra-credit points.) Do problem 39 from chapter 10 of the textbook (p. 659).
8. (Up to 10 extra-credit points.) Do problem 31 from chapter 11 of the textbook (p. 741), but write a complete program rather than only a function. Prompt the user for the required input values (string to insert, string to insert into, position).

9. (Up to 10 extra-credit points.) Do problem 36 from chapter 11 of the textbook (p. 741), but write a complete program rather than only a function. Prompt the user for the required input value (string to be checked for “is it a palindrome?”).
10. (Up to 10 extra-credit points.) Do problem 28 from chapter 12 of the textbook (p. 814), but write a complete program rather than only a function. Prompt the user for the required input values (month, day, year, and days after today — you can use integers for all four).