

# CIS\*2520 Data Structures

Fall 2016

## Assignment 1

(1) (30%) Write a C program that outputs all possible strings, each of which is formed by using the characters 'c', 'a', 'r', 'b', 'o', and 'n' exactly once. Such strings include *carbon*, *carbno*, *carobn*, ..., *nobrac*.

(2) (30%) Ackermann's function  $A(m, n)$  is a two argument function defined as follows:

$$\begin{aligned} A(0, n) &= n + 1 \quad \text{for } n \geq 0 \\ A(m, 0) &= A(m - 1, 1) \quad \text{for } m > 0 \\ A(m, n) &= A(m - 1, A(m, n - 1)) \quad \text{for } m, n > 0 \end{aligned}$$

Write a recursive function that gives the value of Ackermann's function. Test your program, and find out that for what range of integer parameters,  $(m, n)$ , the output of your implementation does not exceed the value of the maximum integer in the C system on the SoCS machines?

(3) (40%) Let  $x$  be a positive real. To calculate the square root of  $x$  by Newton's method, so that the square of the solution differs from  $x$  is within an accuracy of  $\epsilon$ , we start with an initial approximation  $a = x/2$ . If  $|a * a - x| \leq \epsilon$ , we stop with the result  $a$ . Otherwise we replace  $a$  with the next approximation, defined by  $(a + x/a)/2$ . Then, we test the result again. In general, we keep on computing and testing successive approximations until we find one close enough to stop. Write two C functions, using recursion and non-recursion respectively, to implement the above algorithm. You should use a sequence of big real numbers and a small  $\epsilon$  to test your program, and try to find the differences in execution time between the two functions.

**Due time: 18:00, Monday Sept 26, 2016.**