CIS*2520 Data Structures

Fall 2018

Assignment 3 Guidelines

Assignment 3 is due on Monday morning, Nov 5, 2018.

Write a makefile that can be used to generate two executables, same as the one for Assignment 2. The information in *readme* file is the same as before. Other requirements are also the same.

Q1: The program for the first question should take input from the command line, same as the second question in Assignment 2. That is, the entire expression should be in a character string without spaces in it. You will need to write a very simple parser to process the input string. (Parsing a string is to analyze its structure and identify the components.)

You can parse a string by identifying the nested parenthesis pairs, and the three components within each pair. Within a pair, there is a binary operator, and a left operand and a right operand. An operand can be a sub-string in a parenthesis pair, a floating number, or a variable. A pair of parentheses can be identified by counting the opening and closing parentheses included by the pair.

When creating a tree from a string, you may start with the outer most pair of parentheses, identify the operator of it, create the root node for the operator, and then recursively handle the two operands, which may be a sub-string in an inner pairs of parentheses.

To pass an expression as a command line argument, add a `\` before each parenthesis. For example, to pass `(((1.56+4.26)*(x1-2.23))/x2)` to a program, the argument should be `\(\(\(1.56+4.26\)*\(x1-2.23\)\)/x2\)`.

Q2: The 2D array in the second question can be considered as an extension of the 1D array for representing heaps, which we discussed in classroom. A file of sample data is provided for your information.