Homework 5

C++ - Foundations

University of Washington Extension Program

Name_____

 Using the operators new and delete, create a set of C++ functions for constructing dynamic strings. Memory for a dynamic string is to be allocated as necessary. Include functions to concatenate, append, and replace one string with a larger or smaller one. The function *concatenate()* combines two strings while *append ()* adds a character or substring to the end of a named string. For example,

concat: S3 <- S1 + S2

S1: "Bon Jour "

S2: "mon petit frere"

S3: "Bon Jour mon petit frere"

S1, S2 remain unchanged

append: S1 <- S1 + " Ca va, mon ami"

S1: "Bon Jour Ca va, mon ami"

replace: S1 <- S2

S1: "mon petit frere"

S2: "mon petit frere"

Be certain to delete the dynamic instances before exiting your program.

- 2. Write a function *compute()* that takes a pointer to a function that takes two integer arguments and returns an integer. Define 4 arithmetic functions to add, subtract, multiple, and divide two integers. Use the function compute to evaluate each of the arithmetic functions.
- 3. Write a simple program that declares two integers, two pointers to point to those integers, and two pointers to pointers to refer to the pointers.
 - a. Display the values of the integers directly, via the pointers, and via the pointers to pointers.
 - b. Display the values of the pointers and the pointers to pointers. Explain your results and with a simple graphic, illustrate how these pointers may look in memory.

- c. Write a function, swapPointers(int* ptr0, int* ptr1) that takes your two pointers and swaps them. Display the results following the swap in your function and in main. Explain any discrepancies.
- d. Repeat part c with the modified prototype swapPointers(int** ptr0, int** ptr1). Once again, the intention is to swap the original pointers.
- 4. Write a function that requests integers from the user. Use the values to populate column 0 of a 2 dimensional array. For each row, complete the array by entering the square of the element in corresponding position in column 1.
- 5. Use selection sort to sort the array in problem 4 based upon the values in column 0. In selection sort, we first find the largest element in the array and exchange it with the first element's value. We then find the next largest element in the array and exchange it with the second element's value. We continue the process until the array is sorted.