Mauricio Deguchi

UID: 405582562

CS 31

February 16, 2022

**Project 4**

**Array Manipulation Library**

*Challenges:*

The greatest challenge of this project was trying to determine whether each of the functions worked properly. While the previous projects all had large amounts of sample test data from the discussion board, this project not only the output of the function to be correct but also all the manipulations which were made; these manipulations could not be easily checked using <cassert> so greater use of the debugger was required.

My main includes all the test cases but here is at least one test case per function. While some of the early functions made no changes to the array, and thus only their output value was easily checked using assert, the later functions required more inspection. For the later functions, I created another function called *print* which would simply print out all the elements in an array. Through this function, I would be able to print out the array before and after a function made changes to it.

string a[6] = { "136", "135", "135", "gamma", "bEta", "delta" };

string b[4] = { "" , "", "", "uU iI"};

string c[11] = { "samwell", "jon", "margaery", "daenerys", "daenerys", "jon", "jon", "jon", "jOn", "jon", "jon"};

string d[4] = { "aa", "bb", "bb", "cc" };

string e[4] = { "bb", "bb", "dd", "ee" };

string f[7] = { "ee", "dd", "cc", "bb", "aa", "BB", "AA" };

string g[7] = { "AA", "BB", "BB", "BB", "cc", "dd", "ee" };

string folks[8] = { "samwell", "jon", "mArgaEry", "daenerys", "tYriOn", "sansa", "magdalena", "jon" };

string data[5] = { "mama", "mama", "12,", "sansa", "mama" };

// locating minimum tests

assert(locateMinimum(data, 5) == 2);

// common element tests

assert(countCommonElements(f, 7, g, 7) == 7);

// counting vowels test

assert(countVowels(a, 6) == 6);

// has three of given element test

assert(hasThreeOfX(a, 6, "a") == false);

// majority element test

assert(majorityElement(c, 11) == "jon");

//shift right test

assert(shiftRight(data, 5, 2, "foo") == true);

assert(data[0] == "foo"); data[0] = "mama";

assert(data[1] == "foo"); data[1] = "mama";

// rotate left tests

assert(rotateLeft(d, 2, 3) == true);

assert(d[0] == "bb"); d[0] = "aa";

assert(d[1] == "aa"); d[1] = "bb";

assert(d[2] == "bb");

assert(d[3] == "cc");

// replace all tests

assert(replaceAll(folks, 8, 'a', 'Z') == 8);

print(folks, 8);