Report for Project 4

2. a

One obstacle I overcame during this project was figuring out how to sort strings for the splitter method so I wouldn’t have to create a separate array of strings. Another obstacle I faced was keeping track of counters when I was trying to compare strings from different arrays. Also, this is the first time I tried to test my code using assert statements so I had to learn how to use assert statements and the windows local debugger.

b.

int appendToAll(string a[], int n, string value);

string people[5] = { "hillary", "jeb", "rand", "ben", "john" };

appendToAll(people, 5, “!!!”);

appendToAll(people, 0, “”);

appendToAll(people, 3, “s”);

appendToAll(people, -2, “s”);

string people[2] = { "hillary", "hillary" };

appendToAll(people, 5, “!!!”);

appendToAll(people, 0, “”);

appendToAll(people, 3, “s”);

appendToAll(people, -2, “s”);

int lookup(const string a[], int n, string target);

lookup(people, 5, “hillary”);

lookup(people, 3, “jeb”);

lookup(people, 0, “rand”);

lookup(people, -2, “hillary”);

int positionOfMax(const string a[], int n);

positionOfMax(people, 5);

positionOfMax(people, 0);

positionOfMax(people, -3);

int rotateLeft(string a[], int n, int pos);

rotateLeft(people, 5, 2);

rotateLeft(people, 3, 0);

rotateLeft(people, 0, 3);

rotateLeft(people, 0, 0);

rotateLeft(people, -1, -1);

int countRuns(const string a[], int n);

countRuns(people, 3);

countRuns(people, 0);

countRuns(people, 5);

countRuns(people, -2);

int flip(string a[], int n);

flip(people, -2);

flip(people, 0);

flip(people, 3);

flip(people, 5);

int differ(const string a1[], int n1, const string a2[], int n2);

string people[5] = { "hillary", "jeb", "rand", "ben", "john" };

string folks[6] = { "chris", "marco", "", "ben", "donald", "john" };

differ(people, 3, folks, 3);

differ(people, 5, folks, 6);

differ(people, 1, folks, 6);

differ(people, 0, folks, 2);

differ(people, 2, folks, 0);

differ(people, 0, folks, 0);

differ(people, -2, folks, -2);

int subsequence(const string a1[], int n1, const string a2[], int n2);

subsequence(people, 5, folks, 6);

subsequence(people, 3, folks, 3);

subsequence(people, 1, folks, 6);

subsequence(people, 0, folks, 2);

subsequence(people, 2, folks, 0);

subsequence(people, 0, folks, 0);

subsequence(people, -2, folks, -2);

int lookupAny(const string a1[], int n1, const string a2[], int n2);

lookupAny(people, 5, folks, 6);

lookupAny(people, 3, folks, 3);

lookupAny(people, 1, folks, 6);

lookupAny (people, 0, folks, 2);

lookupAny(people, 2, folks, 0);

lookupAny(people, 0, folks, 0);

lookupAny(people, -2, folks, -2);

int split(string a[], int n, string splitter);

split(people, 3, “chess”);

split(people, 5, “chess”);

split(people, 0, “chess”);

split(people, -2, “chess”);

split(people, 4, “”);

split(people, 5, “zz”);