Project 4 Report

I did not encounter many challenges during this project, but the functions that took me the most time to implement were positionOfMax() and divide(). Initially, positionOfMax() looked fairly simple, and my first approach was to use two nested for loops and updating the position of the max. But, the returned value was sometimes incorrect, and it took some time to figure out why. Eventually, I realized only one for loop was necessary and that I overlooked a detail in assigning the max position. For the divide() method, I initially tried to accomplish the task by creating a second array, but after I read the Project 4 spec further and saw the extra credit opportunity, I scrapped my progress and tried using the bubble sort method.

I had 3 arrays and used numerous test cases. The arrays and the test cases are:

string str1 [8] = {“aa”, “bb”, “hh”, “ee”, “cc”, “gg”, “dd”, “ff”}

string str2 [5] = {“aa”, “bb”, “hhh”, “ee”, “cc”}

string str3 [9] = {“a”, “b”, “c”, “c”, “b”, “b”, “b”, “a”, “a"}

1. appendToAll(str1, 0, “.”) — to check how it deals with an array with size 0
2. appendToAll(str2, -1, “ ”) — to check how it deals with an array with a negative size
3. appendToAll(str3, 5, “”) — to check how it deals with an empty string
4. appendToAll(str1, 8, “…”) — to check if it works correctly with the whole array
5. appendToAll(str2, 4, “aa”) — to check if it works correctly with part of the array
6. appendToAll(str1, 4, “??”) — to check if it works correctly with a string of symbols
7. appendToAll(str2, 3, “AA”) — to check if it works correctly with a string of capital letters
8. appendToAll(str3, 7, “bb”) — to check if it works correctly with a string of lowercase letters
9. appendToAll(str1, 5, “cD!”) — to check if it works correctly with a string of mixed symbols
10. lookup(str1, 0, “!!”) — to check how it deals with an array with size 0
11. lookup(str2, -4, “//”) — to check how it deals with an array with a negative size
12. lookup(str3, 7, “”) — to check how it deals with an empty string
13. lookup(str1, 8, “123”) — to check if it works correctly with the whole array
14. lookup(str2, 3, “”) — to check if it works correctly with part of the array
15. lookup(str1, 8, “ee”) — to check if it works correctly with a string that is in the array
16. lookup(str2, 3, “i”) — to check if it works correctly with a string that isn’t in the array
17. positionOfMax(str1, 0) — to check how it deals with an array with size 0
18. positionOfMax(str2, -10) — to check how it deals with an array with a negative size
19. positionOfMax(str1, 8) — to check if it works correctly with the whole array
20. positionOfMax(str2, 2) — to check if it works correctly with part of the array
21. rotateLeft(str1, 0, 0) — to check how it deals with an array with size 0
22. rotateLeft(str2, -4, 1) — to check how it deals with an array with a negative size
23. rotateLeft(str1, 8, 3) — to check if it works correctly with the whole array
24. rotateLeft(str2, 1, 0) — to check if it works correctly with part of the array
25. countRuns(str1, 0) — to check how it deals with an array with size 0
26. countRuns(str2, -7) — to check how it deals with an array with a negative size
27. countRuns(str1, 8) — to check if it works correctly with the whole array
28. countRuns(str2, 4) — to check if it works correctly with part of the array
29. flip(str1, 0) — to check how it deals with an array with size 0
30. flip(str2, -14) — to check how it deals with an array with a negative size
31. flip(str1, 8) — to check if it works correctly with the whole array
32. flip(str2, 2) — to check if it works correctly with part of the array
33. differ(str1, 0, str2, 1) — to check how it deals with an array with size 0
34. differ(str2, 2, str3, 0) — to check how it deals with an array with size 0
35. differ(str1, 0, str3, 0) — to check how it deals with both arrays with size 0
36. differ(str1, -1, str2, 4) — to check how it deals with an array with a negative size
37. differ(str2, 1, str3, -2) — to check how it deals with an array with a negative size
38. differ(str1, -3, str3, -4) — to check how it deals with both arrays with a negative size
39. differ(str1, 8, str2, 2) — to check if it works correctly with one whole array
40. differ(str2, 2, str3, 9) — to check if it works correctly with one whole array
41. differ(str1, 8, str3, 9) — to check how it deals with both whole arrays
42. differ(str1, 3, str3, 6) — to check how it deals with part of both arrays
43. subsequence(str1, 0, str2, 3) — to check how it deals with an array with size 0
44. subsequence(str2, 4, str3, 0) — to check how it deals with an array with size 0
45. subsequence(str1, 0, str3, 0) — to check how it deals with both arrays with size 0
46. subsequence(str1, -4, str2, 2) — to check how it deals with an array with a negative size
47. subsequence(str2, 3, str3, -2) — to check how it deals with an array with a negative size
48. subsequence(str1, -3, str3, -4) — to check how it deals with both arrays with a negative size
49. subsequence(str1, 8, str2, 2) — to check if it works correctly with one whole array
50. subsequence(str2, 2, str3, 9) — to check if it works correctly with one whole array
51. subsequence(str1, 8, str3, 9) — to check how it deals with both whole arrays
52. subsequence(str1, 3, str3, 6) — to check how it deals with part of both arrays
53. lookupAny(str1, 0, str2, 1) — to check how it deals with an array with size 0
54. lookupAny(str2, 2, str3, 0) — to check how it deals with an array with size 0
55. lookupAny(str1, 0, str3, 0) — to check how it deals with both arrays with size 0
56. lookupAny(str1, -1, str2, 4) — to check how it deals with an array with a negative size
57. lookupAny(str2, 1, str3, -2) — to check how it deals with an array with a negative size
58. lookupAny(str1, -3, str3, -4) — to check how it deals with both arrays with a negative size
59. lookupAny(str1, 8, str2, 2) — to check if it works correctly with one whole array
60. lookupAny(str2, 2, str3, 9) — to check if it works correctly with one whole array
61. lookupAny(str1, 8, str3, 9) — to check how it deals with both whole arrays
62. lookupAny(str1, 3, str3, 6) — to check how it deals with part of both arrays
63. divide(str1, 0, “!!”) — to check how it deals with an array with size 0
64. divide(str2, -4, “//”) — to check how it deals with an array with a negative size
65. divide(str3, 7, “”) — to check how it deals with an empty string
66. divide(str1, 8, “123”) — to check if it works correctly with the whole array
67. divide(str2, 3, “”) — to check if it works correctly with part of the array
68. divide(str1, 8, “ee”) — to check if it works correctly with a string that is in the array
69. divide(str2, 3, “i”) — to check if it works correctly with a string that isn’t in the array