

BTE 320 QUIZ ON ARRAYS

Question 1

In this question, you will read words from a file and place them into an array of type string.

- 1- Make a data text file "words.txt" – that contains one word on each line. Use at least 20 words.
- 2- Now write a program that reads the words in the file into an array of strings (a repeated word should not be inserted into the array – your program should not allow that and you should make sure your data file has duplicate words to test this functionality). Make your array size enough to hold 1000 words. To read the words into the array, you should make a function that takes a string array, a data size integer (dsize) by reference and an ifstream – please look at the example we did in class.
- 3- Declare your array and file streams in main and call the function that reads the data into the array.
- 4- Write a printArray function that takes a string array, the dsize and an ostream object so that you could print to the console or to an output file.
- 5- Print your array from main by calling the printArray function – once to the console and once to a file "wordsoutput.txt".
- 6- Use the selectionSort Algorithm that we covered in class to sort the array.
- 7- Repeat #5 and make sure the array is sorted.
- 8- Find the maximum string and the minimum string in the array (remember arrays are compared based on the ASCII value).
- 9- Write a function that takes a string and converts every character of it to uppercase. Now call that function from a function that you pass the array and dsize to, to uppercase all words in the array (convert all words in the array to uppercase letter) and call that from main passing your array to that function. Print the array.

Question 2

In this question, you will read two data files that include integers into two different arrays – the same way we did in class (but we are doing to arrays here). Duplicates are ok.

- 1- After you read the data into the array (use one function that takes an int array and a dsize by reference just like we did in class, and call that from main to fill both arrays).
- 2- Include a printArray function so that you could print the two arrays. Use array size 1000.
- 3- Use a sort algorithm to sort both arrays.
- 4- Now write a compareOrder function that takes two arrays and their data sizes. If the data size is the same and every element in the first array is equal to element in the same corresponding position in the second array (both sorted) – then the arrays are equal (return 0). If the very first element that is different is larger in one array than the other array then the first array is larger (return 1) otherwise (return -1). If the dsizes are different, the longer array is larger (return 1 if array1 is larger than array 2 and -1 otherwise).

- 5- Call the compareOrder function from main passing your two arrays and their corresponding dsizes and give appropriate messages. (Please see Hint - required).
- 6- Now write a compareSum to compare the two arrays based on the sum of their elements – follow the same theme of returning 0,1 or -1 – and call from main with appropriate messaging (please see Hint - required).

Hint: Instead of calling compareOrder and CompareSum and including logic code for messaging in main (ugly), write a void wrapper function compare that would take a compare type, the arrays and their dsizes and will call the appropriate compare function with the arrays and the dsizes and produce the appropriate messaging).

Submission

- 1- **Submit your cpp files pasted in a word document. Paste the code not an image.**
- 2- **Submit the data files you used separately as files not pasted in the word document.**
- 3- **Include a screenshot of every output for each case. No screenshots = ZERO. You must paste the screenshot with the appropriate question in the word document – I will not accept image files. You need to follow instructions of how to submit.**
- 4- **For Question 2, please make different sets of two data files to test your code for all cases.**