

Assignment # 4

Programming fundamentals

4th December, 2020 11:59pm

Weightage 5.

Note.

- Please avoid plagiarism.
- Contact me on email in case of any query.
- No late submissions would be accepted.
- Submit assignment on google classroom and note google classroom also checks plagiarism, so be careful.
- Only 25 to 30% plagiarism is acceptable.

Q1. Write the two functions, one for sorting the numeric array and one for sorting the character arrays/strings.

```
sortAlphabeticalArray (char array[], int len);
```

```
sortNumericArray ( int array[], int len);
```

Q2. Try to solve the question 1 with the help of the recursion.

Q3. Implement the functions strlen, strcat, strcpy and strncpy. There should be two implementations of this one should include call by reference and other should be call by value.

Q4. Implement the following algorithm in c.

```
quicksort (array){
    if (array.length > 1){
        choose a pivot;
        while (there are items left in array){
            if (item < pivot)
                put item into subarray1;
            else
                put item into subarray2;
        }
        quicksort(subarray1);
        quicksort(subarray2);
    }
}
```

Q5. Describe the importance of recursion in the above algorithm. Can you depict the whole process in the pictorial way by taking any example of your own choice? (Hint: You need to show how the array would be sorted using this).

Q6. Implement the following pseudo code using C.

Binary search recursion: pseudo-code

```
// Refer to BinarySearchApp project

Boolean BS(A, key, start, end)
    mid = (start+end)/2
    if (A[mid] == key)
        return true
    else
        if (end <= start)
            return false
        else
            if (A[mid] > key)
                return BS(A, key, start, mid-1)
            else
                return BS(A, key, mid+1, end)
```



Q7. In class we discussed about Stack and queue. Can you give real time examples of the problems which can be implemented using the queue and the stack?

Q8. Draw any two shapes of your choice using recursion.