Assignment 2 - DSA - UEC613

Ishaan Bhola - 102015051 - 2NC6

Q1) If a one dimensional integer type array with its size and length given, write the code in C++ language to create functions to perform the following operations (Please assume whatever is necessary to exemplify the results) :

i) Display()
Function:-

```
    private static void display(int[] arr) {
        for(int val:arr){
            System.out.print(val+" ");
        }
        System.out.println();
    }
```

Output:-
Array = {1,2,3,4,5}

```
1 2 3 4 5
```

ii) Add/Append(x)
Function:-

```
    private static int[] add(int arr[],int x){
        int newArr[]=new int[arr.length+1];
        for(int i=0;i<arr.length;i++){
            newArr[i]=arr[i];
        }
        newArr[arr.length]=x;
        return newArr;
    }
```

Output:-
Adding 10 to the first array

```
1 2 3 4 5
1 2 3 4 5 10
```

iii) Insert(index,x)
Function:-

```
private static int[] insert(int arr[],int index,int element){
    int newArr[]=new int[arr.length+1];
    int i;
    for(i=0;i<index;i++){
        newArr[i]=arr[i];
    }
    newArr[i]=element;
    for(i=index+1;i<newArr.length;i++){
        newArr[i]=arr[i-1];
    }
    return newArr;
}
```

Output:-
Inserting 10 at index 1 of the first array

```
1 2 3 4 5
1 10 2 3 4 5
```

iv)Delete(x)
Function:-

```
private static int[] delete(int arr[],int index){
    int newArr[]=new int[arr.length-1];
    for(int i=0;i<index;i++){
        newArr[i]=arr[i];
    }
    for(int i=index;i<newArr.length;i++){
        newArr[i]=arr[i+1];
    }
    return newArr;
}
```

Output:-
Deleting element at index 1 of first array

```
1 2 3 4 5
1 3 4 5
```

v) LinearSearch(s)
Function:-

```
    private static boolean linearSearch(int x,int arr[]){
        for(int val:arr){
            if(val==x) return true;
        }
        return false;
    }
```

Output:-
Checked if 2 exists in {1,2,3,4,5}

```
true
```

vi)Get(index); function to get value available on the given index
Function:-

```
    private static int get(int index,int arr[]){
        return arr[index];
    }
```

Output:-
Array = {1,2,3,4,5}, obtained 4th element

```
5
```

vii)Set(index,x); insert the value x at the given index
Function:-

```
    private static void set(int index,int x,int arr[]){
        arr[index]=x;
    }
```

Output:-
None as function is void

viii)Max()
Function:-

```
    private static int max(int arr[]){
        int max = Integer.MIN_VALUE;
        for(int val:arr){
            max=Math.max(max, val);
        }
        return max;
    }
```

Output:-
Max in array = {1,2,3,4,5}

```
5
```

ix) Min()
Function:-

```
    private static int min(int arr[]){
        int min = Integer.MAX_VALUE;
        for(int val:arr){
            min=Math.min(min, val);
        }
        return min;
    }
```

Output:-
Min in array = {1,2,3,4,5}

```
1
```

x)Reverse()
Function:-

```
    private static void reverse(int arr[]){
        int i=0,j=arr.length-1;
        while(i<j){
            int temp = arr[i];
            arr[i]=arr[j];
            arr[j]=temp;
            i++;
            j--;
        }
    }
```

Output:-
Reversed the first array

```
1 2 3 4 5
5 4 3 2 1
```

xi)Shift()
Function:-

```
    private static void shift(int arr[]){
        int next = arr[0];
        for(int i=1;i<arr.length;i++){
            int temp = arr[i];
            arr[i]=next;
            next=temp;
        }
        arr[0]=next;
    }
```

Output:-
Shifted the first array

```
1 2 3 4 5
5 1 2 3 4
```

xii) Rotate(int number)

Function:-

```
    private static void rotate(int arr[],int x){
        for(int i=0;i<x;i++){
            shift(arr);
        }
    }
```

Output:-

Rotated the first array by 3

```
1 2 3 4 5
3 4 5 1 2
```

Q2) For a given array, write functions to perform the following:

i. Check if an array is sorted
Function:-

```
    private static boolean isSorted(int arr[]) {
        for(int i=1;i<arr.length;i++){
            if(arr[i-1] > arr[i]) return false;
        }
        return true;
    }
```

Output:-
Array={1,2,3,4,5}

```
true
```

ii. Merge arrays
Function:-

```
    private static int[] mergeArrays(int a[],int b[]) {
        int n=a.length+b.length;
        int c[]=new int[n];

        for(int i=0;i<a.length;i++){
            c[i]=a[i];
        }
        for(int i=0;i<b.length;i++){
            c[i]=b[i];
        }
        return c;
    }
```

Output:-
Merging the first and second arrays

```
1 2 3 4 5
6 7 8 9 10
1 2 3 4 5 6 7 8 9 10
```

Q3) For a given array, write functions to perform the following:
i. Finding single element in an array
Function:-

```
private static int findSingleElement(int arr[],int element){
    for(int i=0;i<arr.length;i++){
        if(arr[i]==element) return i;
    }
    return -1;
}
```

Output:-
Returns the index of element 5 in {1,2,3,4,5}

```
4
```

iii. Finding duplicates in a sorted array
Function:-

```
private static int findDuplicatesInSorted(int arr[]){
    for(int i=1;i<arr.length;i++){
        if(arr[i-1]==arr[i]) return arr[i];
    }
    return -1;
}
```

Output:-
Returns the duplicate element in {1,2,3,4,5}

```
2
```

iv. Finding duplicates in an unsorted array;
Function:-

```
    private static int findDuplicatesInUnSorted(int arr[]){
        for(int i=0;i<arr.length;i++){
            for(int j=i+1;j<arr.length;j++){
                if(i!=j && arr[i]==arr[j]) return arr[i];
            }
        }
        return -1;
    }
```

Output:-
Returns the duplicate element in {5,4,3,2,2,6}

```
2
```

v. Finding a pair of elements with sum k

```
    private static int[] pairWithSumK(int arr[],int k){
        for(int i=0;i<arr.length;i++){
            for(int j=i+1;j<arr.length;j++){
                if(i!=j && arr[i]+arr[j]==k) return new
int[]{arr[i],arr[j]};
            }
        }
        return new int[]{};
    }
```

Output:-
Output for array={1,2,3,4,5} where the sum of the pair should be 6

```
1 5
```

vi. Finding max and min in a single scan; here you should use only single loop to perform both the operations

```
    private static int[] maxMin(int arr[]){
        int min=arr[0],max=arr[0];
        for(int i=1;i<arr.length;i++){
            max = Math.max(arr[i], max);
            min = Math.min(arr[i],min);
        }
        int sol[]={min,max};
        return sol;
    }
```

Output:-
Array={1,2,3,4,5}
Min and Max respectively

```
1 5
```