

HW2

Due: 11:00 pm, 10/13/2021

Instructor: Jun Yi

1) (50 points) In this problem, we will write a program that, given an integer k , an integer n , and a list of integers, outputs the number of pairs in the list that add to k . To receive full credit for design, your algorithm must have a runtime in $O(n)$, where n is the length of the list of integers. Your program should take as input a number k on its own line, followed by a number n on its own line, followed by a space-separated list of n integers. It should then output the number of pairs in the list that add to k . Order does not matter when considering a pair (in other words, in the list $[2,1]$ there is one distinct pair that sums to 3, not two). You may assume that all numbers in the input are distinct.

For example, if our `_le input.txt` is:

```
1
6
```

```
1 2 3 4 -2 -3
```

then we should print:

```
2
```

since there are two pairs that sum to 1: $3+(-2)$ and $4+(-3)$. As another example, if input array is:

```
3
4
```

```
1 2 3 4
```

then we should print:

```
1
```

since there is one pair that sums to 3: $1+2$

After you finish your code, please test

```
10
9
```

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

2) (30 points) Given three arrays `arr1`, `arr2` and `arr3`, the elements of `arr2` are distinct, and all elements in `arr2` are also in `arr1`. Meanwhile, all elements in `arr2` are also in `arr3`.

Sort the elements of arr1 such that the relative ordering of items in arr1 are the same as in arr2. Elements that do not appear in arr2 should be placed at the end of arr1 in ascending order. Meanwhile, Elements for arr3 that do not appear in arr2 should be placed at the front of arr1 in ascending order. For example,

arr1:[2,4,3,5,6,1,9]

arr2: [2,4,3]

arr3: [2,4,3,11,8,13]

the output should be [8,11,13,2,4,3,1,5,6,9] since [2,4,3] shared by arr1, arr2,arr3. Elements [11,8,13] should have a ascending arder in front of arr1, and elements [5,6,1,9] have an ascending order at the end of arr1.

You can define your input arrays.

3) (20) Given an integer array nums, move all the even integers at the beginning of the array followed by all the odd integers. Both even and odd numbers should follow an ascending order in the sorted array.

Return the sorted array

For example,

input: [1,4,2,3]

output: [2,4,1,3]

You can defined your input array.