# Data Structures and Algorithms (CS F211)
## Second Semester 2018-2019
## Lab Sheet 4

**1.** You go to the Egyptian market to get souvenirs for your friends and family. The market has weird rules. It has *n* different items numbered from *1* to *n*. Each item *i* has a **base cost** of *a[i]* Egyptian pounds. If you buy *k* different items with indices $x_1, x_2, x_3, ..., x_k$ then the cost of each item is *a[i]+x$_j$*k* where *j* varies from 1 to *k*. In other words, the cost of an item is equal to its base cost in addition to its index multiplied by the factor *k*.

You want to buy as many souvenirs as possible and your budget is *S* Egyptian dollars. Find the maximum number of items you can buy. In case the number of items is same, choose the one with the lowest total cost.

**Input:**
First line contains *n* and your total budget *S*.
Second line contains *n* different integers each of which denotes the cost of each item.
**Output:**
Two integers which denote the number of items and total cost you will be spending.
**Example:**
3 14
2 3 5
**Answer:**
2 11
**Explanation:**
You cannot get 3 items because they will cost you [5,9,14] and the total cost would be 28. If you get 2 items, then the items would cost [4,7]. So, you can get items costing 4,7 which will cost you a total of 11 Egyptian dollars.

**2.** You are trying to solve questions for the DSA lab exam. The question set consists of a large number of different questions. You are exhausted today, so you decide to solve the problems like this:
You first solve *v* problems, the drink a cup of tea and solve floor(*v/k*) problems (where floor() function returns the largest possible integer value which is less than or equal to the given argument), then floor(*v/(k*k)* ), then floor(*v/k*k*k*) and so on…
The moment the value floor(*v/(k^p)* ) is equal to zero you stop and go to sleep.
Find the minimum value of *v* such that you solve not less than *n* questions.

**Input:**
Two integers *n* and *k*.
**Output:**
Print only one integer – the minimum value of *v* that lets you solve *n* questions.
**Example:**
7 2
**Answer:**
4
**Explanation:**
When *v* = 4, then he solves the questions like this: 4, 2, 1.
Thus, he manages to solve 4+2+1 = 7 questions.

**3.** You are given a sorted list of elements. All of the elements in the array occur twice except for one element. Find the element that occurs only once.

**Example:**
1 1 4 4 5 6 6
**Answer:**
5

**4.** You and your friends are playing the game of stacking a pile of dominos. There are several towers of dominos each with some height. You are given with some initial configuration of the towers. To increase/decrease the height you need to add/remove the dominos. The number of dominos in the game remains the same. Adding/removing incurs a cost which is different for different towers. You are assigned with the task of making a set of towers such that all the towers have same height and the cost incurred is as minimum as possible. Note that the initial number of towers and final number of towers may not be the same.

**Input:**
First line contains the number of towers.
Second line contains the number of dominos in each tower.
Third line contains the cost incurred for each tower when adding/removing a domino.
**Output:**
Minimum cost, number of towers, height of each tower
**Example:**
3
1 2 3
10 100 200
**Answer:**
110, 2, 3

**Explanation:** Remove a domino from Tower 1 (Cost 10) and add it to Tower 2 (Cost 100). Total cost is 110. Note that the number of dominos (6) remains the same, however, the number of towers has become 2 and the height of each tower is 3.

**5.** You are working in a factory and you have *n* machines running simultaneously. Each of the machines takes a certain time (in secs) to produce an item. The time is different for different machines. Your task is to find the minimum time required to produce *m* items using Binary Search.

**Input:** First line contains *n* and *m*.
Second line contains an array which denotes how much each machine takes to produce an item.
**Output:**
Single integer denoting the minimum time to produce *m* items.
**Example:**
3 11
1 2 3
**Answer:**
6
**Explanation:**
In 6 secs, machine 1 will produce 6 items, machine 2 will produce 3 and machine 3 will produce 2.

**6.** Tony stark is lost in space and need to record his final message on a camera. As he was searching for the camera he found a secret scroll by Dr. Strange that had valuable information. The scroll key is a sum of two numbers whose absolute difference is equal to a number **d** on the scroll. Furthermore, those numbers can only be taken from a given set of **n** numbers on the scroll. If there is no key possible return the answer as -1 to Stark otherwise find the key. Solve this using binary search.

**Constraints:**
1<=n<=100000
1<=d<=10000
1<=Number on Scroll<=10000

**Input:**
This line contains integer **n**
This line contains the array of integers on the scroll
This line contains the absolute difference **d**

**Sample Input:**
6
5 20 3 2 50 80
78

**Sample Output:**
82

**Sample Input:**
6
5 220 13 122 550 80
23

**Sample Output:**
-1

**7.** Senti and Panda are best friends. Senti loves cocktails. Senti has decided to rank his favourite **n** cocktails based on their price. Panda as a prank decided to swap two adjacent numbers on the ranked order. Senti got very emotional after that and asked his friend Moggie to order a drink for him for **k** price and also tell him its current rank on the new list. Help Moggie to find the current rank. If it is not there on the list return -1. Solve this problem using binary search.

**Constraints:**
1<=n<=100000
1<=k<=10000
1<=Price of a cocktail<=10000

**Input:**
This line contains integer n
This line contains an updated price list
This line contains k

**Sample Input:**
7
3  10 40 20 50 70 80
40

**Sample Output:**
3

**Sample Input:**
6
51 221 131 1222 5550 8000
123

**Sample Output:**
-1


**8.** Jon Snow likes Ygritte and decided to propose her. As Ygritte is very shrewd, she decided to give him a task. She gives him a sorted list of **n** numbers and rotates it several times. She now asked him to find the position of number **k** on the list as fast as possible. Help him to find the love of his life. Solve this problem using binary search. Assume that index starts from 1.

**Constraints:**
1<=n<=100000
1<=k<=10000
1<=number on the list<=10000

**Input:**
This line contains integer **n**
This line contains numbers on the list
This line contains **k**

**Sam ple Input:**
5
4 5 1 2 3
3

**Sample Output:**
5

**Sample Input:**
5
2 4 6 0 1
6

**Sample Output:**
3


**8.** Naruto and Sasuke are on a mission of finding number **k** among **n** numbers or inserting them. Naruto does the job of finding the number and if it is not there then Sasuke inserts it at the apt position. They need to do it as quickly as possible. Help them to solve in O(**log n**) time using binary search. Also, provide the name of the person who did the last action. Assume that index starts from 1.

**Constraints:**
1<=**n**<=100000

1<=number on the list<=10000
1<=***k***<=10000

**Input:**
This line contains integer ***n***
This line contains numbers on the list
This line contains ***k***

**Sample Input:**
4
1 3 5 6
5

**Sample Output:**
Naruto 3

**Sample Input:**
4
1 3 5 6
2

**Sample Output:**
Sasuke 2

**9.** Kevin Durant is trying to play the game of odd numbers. He is being given an array of ***n*** numbers where all elements appear even number of times except one. All repeating occurrences of elements appear in pairs and these pairs are not adjacent (there cannot be more than two consecutive occurrences of any element). Find the element that appears an odd number of times.

**Constraints:**
1<=n<=100000
1<=number on the list<=10000

**Input:**
This line contains integer n
This line contains numbers on the list

**Sample Input:**
15
1 1 2 2 1 1 2 2 13 1 1 40 40 13 13

**Sample Output:**
13

**Sample Input:**
13
1 1 2 2 3 3 4 4 3 600 600 4 4

**Sample Output:**
3

************************