**CODE N RUN**

**Q1.** Tom is a string freak. He has got sequences of  n words to manipulate. If in a sequence, two same words come together then he’ll destroy each other. He wants to know the number of words left in the sequence after this pairwise destruction.

**Input:**  
The first line of input contains an integer denoting the no of test cases. Then T test cases follow. Each test case contains two lines. The first line of input contains an integer n denoting the number of words in a sequence. In the second line are n space separated words of the sequence. Words are contiguous stretches of printable characters delimited by white space.

**Output:**  
For each test case in a new line print the number of words left per sequence.

**Constraints:**  
1<=T<=100  
1<=n<=100

**Example:  
Input:**  
2  
5  
ab aa aa bcd ab  
4  
tom jerry jerry tom

**Output:**  
3  
0

**Q 2**.**GIVE THE CHOCOLATES!**

Given an array of n integers where each value represents number of chocolates in a packet. Each packet can have variable number of chocolates. There are m students, the task is to distribute chocolate packets such that:

1. Each student gets one packet.

2. The difference between the number of chocolates in packet with maximum

chocolates and packet with minimum chocolates given to the students is minimum.

Input Format : An integer n which is the size of the array and then ‘n’ array elements , an integer m denoting the number of students .

**Output Format : A number indicating minimum difference .**

**Sample Test 1** : Input : n= 7 arr[] = {7, 3, 2, 4, 9, 12, 56} m = 3

Output: Minimum Difference is 2

We have seven packets of chocolates and we need to pick three packets for 3 students If we pick 2, 3 and 4, we get the minimum difference between maximum and minimum packet sizes.

**Sample Test 2:** Input : n = 8 arr[] = {3, 4, 1, 9, 56, 7, 9, 12} m = 5

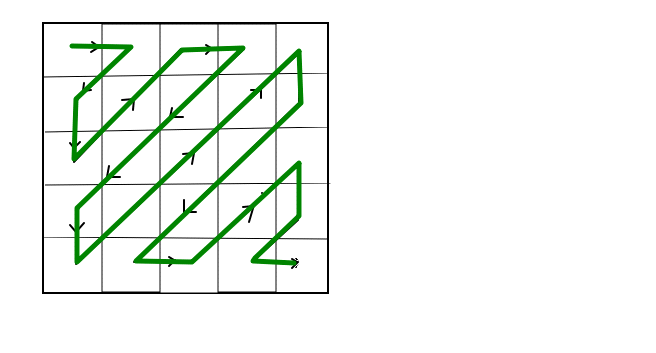
Output: Minimum Difference is 6 The set goes like 3,4,7,9,9 and the output is 9-3 = 6

**Sample Test 3**: Input : n = 17 arr[] = {12, 4, 7, 9, 2, 23, 25, 41,30, 40, 28, 42, 30, 44, 48, 43, 50} m = 7 Output: Minimum Difference is 10 We need to pick 7 packets. We pick 40, 41, 42, 44, 48, 43 and 50 to minimize difference between maximum and minimum.

**Q3.**Given a binary string, count number of substrings(IT’S NOT WRITTEN DISTINCT) that start and end with 1. For example, if the input string is “00100101”, then there are three substrings “1001”, “100101” and “101”.

Input Format : A String containing 0’s and 1’s.

Output Format : A number indicating the number of strings starting and ending with 1.

**Q4**.Given a matrix of 2D array of n rows and m coloumns. Print this matrix in ZIG-ZAG fashion as shown in figure.  


**Example:**

Input:

1 2 3

4 5 6

7 8 9

Output:

1 2 4 7 5 3 6 8 9

**Q5.** Find if there is a rectangle in binary matrix with corners as 1

There is a given binary matrix, we need to find if there exists any rectangle or square in the given matrix whose all four corners are equal to 1.

Examples:

Input :

mat[][] = { 1 0 0 1 0

0 0 1 0 1

0 0 0 1 0

1 0 1 0 1}

Output : Yes

as there exists-

1 0 1

0 1 0

1 0 1  
**Output:**

Yes