

애너그램 수

Description

애너그램(anagram)이란 알파벳이 나열된 순서는 다르지만, 알파벳의 구성이 일치하는 문자열을 말한다.

예를 들어, cat의 애너그램은 cta, act, atc, tca, tac가 있다.

애너그램 수는 어떤 양의 정수 N에 대하여 각 자릿수의 구성이 일치하는 수를 말한다.

예를 들어, 123의 애너그램 수는 132, 213, 231, 312, 321이 있다.

양의 정수 N을 입력으로 받아, 주어진 양의 정수들 중에서 N과 애너그램인 수가 모두 몇 개인지 출력하시오.

단, N의 각 자릿수는 0은 포함하지 않는다고 가정해도 된다.

애너그램을 판별하는 가장 쉬운 방법은 정렬한 후에 두 문자열이 같은 지를 판별하는 것이다.

다음 프로그램에서 버블 정렬을 이용하여 scan(), comp() 함수의 구현을 완성하여 제출하시오.

```
#include <stdio.h>
#define MAX 10

void swap(int a[], int i, int j);
void bubblesort(int n, int a[]);

/* converts decimal digits into array a[] */
int scan(int n, int a[]);

/* returns 1 if all the elements of a[] and b[]
 * are equal in the range from 0 to n - 1.
 * returns 0, otherwise.
 */
int comp(int n, int a[], int b[]);

int main()
{
    int n, t, m, a[MAX], b[MAX], cnt = 0;
    scanf("%d", &n);
    int len = scan(n, a);
    bubblesort(len, a);
    scanf("%d", &t);
    for (int i = 0; i < t; i++) {
        scanf("%d", &m);
        int lenm = scan(m, b);
        bubblesort(lenm, b);
        if (len == lenm && comp(len, a, b))
            cnt++;
    }
    printf("%d", cnt);
}
```

Input

첫 번째 줄에 양의 정수 N이 주어진다.

N의 각 자릿수는 0을 포함하지 않으며, 각 자릿수는 모두 서로 다른 숫자이다.

두 번째 줄에 양의 정수 M이 주어진다.

세 번째 줄에 M개의 양의 정수가 공백으로 구분되어 주어진다.

주어지는 모든 수들의 자릿수는 N과 동일하며, N과 마찬가지로 0을 포함하지 않으며 , 각 자릿수는 모두 다른 숫자이다.

Output

첫 번째 줄에 N과 애너그램인 수의 개수를 출력한다.

Sample Input 1

123
9
123 132 213 231 312 321 234 567 891

Sample Output 1

6

Sample Input 2

1248
11
9173 8265 1234 5286 4812 2481 8214 1428 4218 4182 7312

Sample Output 2

6

```
1 int comp(int n, int a[], int b[])
2 {
3     for (int x = 0; x < n; x++)
4     {
5         if (a[x] != b[x])
6         {
7             return 0;
8         }
9     }
10
11     return 1;
12 }
13
14 int scan(int n, int a[])
15 {
16     // 배열에 한 숫자씩 넣고 길이를 출력합니다
17     int count = 0;
18     // int a[] = num을 넣는 곳입니다.
19     while (n != 0)
20     {
21         int temp = n % 10;
22         a[count] = temp;
23         n /= 10;
24         count++;
25     }
26     return count;
27 }
28
29 void bubblesort(int n, int a[])
30 {
31     for (int i = 0; i < n - 1; i++) // n값이 배열 사이즈 기준으로 주어지는지 아니면 일반적 상식 기준에서 크기로 주어지는 지 따라서 코드 수정할 것.
32     {
33         for (int j = 0; j < n - i - 1; j++)
34         {
35             if (a[j] > a[j + 1])
36             {
37                 // printf("a[j], a[j + 1] is %d %d\n", a[j], a[j + 1]);
38                 swap(a, j, j + 1);
39             }
40         }
41     }
42 }
43
44 void swap(int a[], int i, int j)
45 {
46     int temp;
47     temp = a[i];
48     a[i] = a[j];
49     a[j] = temp;
```