GE23131-Programming Using C-2024

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Monday, 23 December 2024, 2:50 PM
Duration	2 hours 43 mins

Question **1**
Correct
Marked out of 1.00

Two strings ***A*** and ***B*** comprising of lower case English letters are compatible if they are equal or can be made equal by following this step any number of times:

- Select a prefix from the string ***A*** (possibly empty), and increase the alphabetical value of all the characters in the prefix by the same valid amount. For example, if the string is ***xyz*** and we select the prefix ***xy*** then we can convert it to ***yx*** by increasing the alphabetical value by 1. But if we select the prefix ***xyz*** then we cannot increase the alphabetical value.

Your task is to determine if given strings ***A*** and ***B*** are compatible.

Input format

First line: String ***A***

Next line: String ***B***

Output format

For each test case, print ***YES*** if string ***A*** can be converted to string ***B***, otherwise print ***NO***.

Constraints

1 ≤ len(A) ≤ 1000000

1 ≤ len(B) ≤ 1000000

SAMPLE INPUT

abaca

cdbda

SAMPLE OUTPUT

YES

Explanation

The string ***abaca*** can be converted to ***bcbda*** in one move and to ***cdbda*** in the next move.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

int main() {
    char A[1000001], B[1000001];
    int i, lenA, lenB, flag = 1;

    scanf("%s", A);
    scanf("%s", B);

    lenA = strlen(A);
    lenB = strlen(B);

    if (lenA != lenB) {
        printf("NO\n");
        return 0;
    }

    for (i = 0; i < lenA; i++) {
        if (A[i] > B[i]) {
            flag = 0;
            break;
        }
    }

    if (flag) {
        printf("YES\n");
    } else {
        printf("NO\n");
    }

    return 0;
}

```

	Input	Expected	Got	
✓	abaca cdbda	YES	YES	✓

Passed all tests! ✓

Question **2**
Correct
Marked out of 1.00

Danny has a possible list of passwords of Manny's facebook account. All passwords length is odd. But Danny knows that Manny is a big fan of palindromes. So, his password and reverse of his password both should be in the list.

You have to print the length of Manny's password and it's middle character.

Note: The solution will be unique.

INPUT

The first line of input contains the integer N, the number of possible passwords.

Each of the following N lines contains a single word, its length being an odd number greater than 2 and lesser than ***14***. All characters are lowercase letters of the English alphabet.

OUTPUT

The first and only line of output must contain the length of the correct password and its central letter.

CONSTRAINTS

1 ≤ N ≤ 100

SAMPLE INPUT

4

abc

def

feg

cba

SAMPLE OUTPUT

3 b

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>


void reverseString(char *str, char *reversed) {
    int len = strlen(str);
    for (int i = 0; i < len; i++) {
        reversed[i] = str[len - 1 - i];
    }
    reversed[len] = '\0';
}

int main() {
    int N;
    scanf("%d", &N);

    char passwords[100][14];
    for (int i = 0; i < N; i++) {
        scanf("%s", passwords[i]);
    }

    char reversed[14];
    for (int i = 0; i < N; i++) {
        reverseString(passwords[i], reversed);
        for (int j = 0; j < N; j++) {
            if (strcmp(passwords[j], reversed) == 0) {
                int len = strlen(passwords[i]);
                printf("%d %c\n", len, passwords[i][len / 2]);
                return 0;
            }
        }
    }

    return 0;
}
```

	Input	Expected	Got	
✓	4 abc def feg cba	3 b	3 b	✓

Passed all tests! ✓

Question **3**
Correct
Marked out of 1.00

Joey loves to eat Pizza. But he is worried as the quality of pizza made by most of the restaurants is deteriorating. The last few pizzas ordered by him did not taste good :(. Joey is feeling extremely hungry and wants to eat pizza. But he is confused about the restaurant from where he should order. As always he asks Chandler for help.

Chandler suggests that Joey should give each restaurant some points, and then choose the restaurant having **maximum points**. If more than one restaurant has same points, Joey can choose the one with **lexicographically smallest** name.

Joey has assigned points to all the restaurants, but can't figure out which restaurant satisfies Chandler's criteria. Can you help him out?

Input:

First line has N, the total number of restaurants.

Next N lines contain Name of Restaurant and Points awarded by Joey, separated by a space. Restaurant name has **no spaces**, all lowercase letters and will not be more than 20 characters.

Output:

Print the name of the restaurant that Joey should choose.

Constraints:

1 <= N <= 10^5

1 <= Points <= 10^6

SAMPLE INPUT

3

Pizzeria 108

Dominos 145

Pizzapizza 49

SAMPLE OUTPUT

Dominos

Explanation

Dominos has maximum points.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

#define MAX_RESTAURANTS 100005
#define MAX_NAME_LENGTH 21

int main() {
    int n, i, maxPoints = 0;
    char bestRestaurant[MAX_NAME_LENGTH], name[MAX_NAME_LENGTH];
    int points;

    scanf("%d", &n);

    for (i = 0; i < n; i++) {
        scanf("%s %d", name, &points);

        if (points > maxPoints || (points == maxPoints && strcmp(name, bestRestaurant) < 0)) {
            maxPoints = points;
            strcpy(bestRestaurant, name);
        }
    }

    printf("%s\n", bestRestaurant);

    return 0;
}
```

	Input	Expected	Got	
✓	3 Pizzeria 108 Dominos 145 Pizzapizza 49	Dominos	Dominos	✓

Passed all tests! ✓

Question **4**
Correct
Marked out of 1.00

These days Bechan Chacha is depressed because his crush gave him list of mobile number some of them are valid and some of them are invalid. Bechan Chacha has special power that he can pick his crush number only if he has valid set of mobile numbers. Help him to determine the valid numbers.

You are given a string "S" and you have to determine whether it is Valid mobile number or not. Mobile number is valid only if it is of length 10 , consists of numeric values and it shouldn't have prefix zeroes.

Input:

First line of input is T representing total number of test cases.

Next T line each representing "S" as described in in problem statement.

Output:

Print "YES" if it is valid mobile number else print "NO".

Note: Quotes are for clarity.

Constraints:

1<= T <= 10^3

sum of string length <= 10^5

SAMPLE INPUT

3

1234567890

0123456789

0123456.87

SAMPLE OUTPUT

YES

NO

NO

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

int main() {
    int T;
    scanf("%d", &T);

    while (T--) {
        char S[100];
        scanf("%s", S);

        int len = strlen(S);
        int isValid = 1;

        if (len != 10) {
            isValid = 0;
        } else if (S[0] == '0') {
            isValid = 0;
        } else {
            for (int i = 0; i < len; i++) {
                if (S[i] < '0' || S[i] > '9') {
                    isValid = 0;
                    break;
                }
            }
        }

        if (isValid) {
            printf("YES\n");
        } else {
            printf("NO\n");
        }
    }

    return 0;
}
```

	Input	Expected	Got	
✓	3 1234567890 0123456789 0123456.87	YES NO NO	YES NO NO	✓

Passed all tests! ✓