Task:9

string processing. Basics, string functions (loaded on different language), string operations, Two algorithms for string pattern matching - Naive approach and Robin K approach.
merge two strings.

Given two strings, s_1 and s_2 as input. The task is to merge them alternatively i.e. the first character of s_1,

Input:
The first line of input contains an integer T denoting the number of test cases. Then T test cases follow each test case contain two strings

Output:
for each test case, in a new line, print the merged string.

Constraints:

$1 <= T <= 100$

$1 <= |s_1|, |s_2| <= 10^4$

Input:

2
Hello
Bye abcdef

output:

HBeyeleLo

abcdadbecf

algorithm:

a) read the first string S_1

b) read the second string S_2

c) Initialize two variable i and j too

d) initialize an empty string merged str

(i) append the i-th character of S_1 to merged str

(ii) append the j-th character of S_2 to merged str

(iii) increment i by 1

(iv) increment j by 1

e) if i is less than the length of S_1, append remaining characters.

f) if j is less than the length of S_2, append remaining characters.

g) print merged str.

Program:

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

```
int main()
{
int t;
scanf("%d",&t);
while(t--)
{
char s1[10005], s2[10005] ans[20005];
scanf("%s %s",s1,s2);
int len1 = strlen(s1), len2 = strlen(s2);
int i,j,k; i=j=k=0;
alternately while(i<len1 && j<len2)
{
ans[k++] = s1[i++];
ans[k++] = s2[j++];
}
while(i<len1) ans[k++] = s1[i++];
while(j<len2) ans[k++] = s2[j++];
ans[k] = '\0';
printf("%s\n", ans);
}
return 0;
}
```

Taking Input:

You are given two inputs: a(integer) and b(string). you need to take the input and print a and b seperated by a space.

Output:

for each test case, print a and b seperated by a space. your task:

This is a function problem. you need to write the function problem to take input a and b inside the function input data()

Constraints:

$1 <= T <= 10$

$1 <= a <= 100$

Input:

2

5

Hello

7

Geeks

Output:

5 Hello

7 Geek

Algorithm:

1. Read the no. of test cases T from the input
2. Loop through T Test cases
   a. Read the integer a from input
   b. Read the string b from input using fgets() or scanf("%[^\n]s",b)
3. end loop

Program:

```
#include <iostream> using
name space std;
void input Data()
{
 int a;
 string b;
 cin >> t;
 while(t--)
 {
  cin >> a >> b;
  cout << a << " " << b << endl
 }
}
int main()
{
 input Data();
 return 0
}
```

output:

2
5
Hello
7
Geeks
5 Hello
7 Geeks

VEL TECH - CSE	
EX.NO	9
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	
TOTAL (15)	14
SIGN WITH DATE	

Result: Thus the program is executed and verified successfully.