

# Praktikum

---

## *String, StringBuffer dan StringBuilder*

---

### Soal String Dasar

1. Terdapat sebuah kalimat dan sebuah karakter yang akan dicari pada kalimat tersebut. Tampilkan output karakter tersebut pada indeks berapa saja.

Contoh :

**Input.**

Kalimat = "abcdefghijklaml".

Karakter = 'a' ;

**Output.**

Karakter 'a' berada di indeks 0, 10, 14.

2. Terdapat sebuah kalimat dan pola yang akan dicari pada kalimat tersebut, carilah pola tersebut di awal dan diakhir kalimat.

**Input.**

Kalimat = "abcdefghijklabcml".

pola = "abc" ;

**Output.**

Pola "abc" diawal kalimat pada indeks 0.

Pola "abc" diakhir kalimat pada indeks 16.

3. Terdapat sebuah kalimat dan pola yang akan dicari pada kalimat tersebut, hitung berapa jumlah pola yang ada pada kalimat.

**Input.**

Kalimat = "abcdefghijklabcml".

pola = "abc" ;

**Output.**

Pola "abc" berada di indeks 0, 10, 16.

4. Pecahlah kalimat menjadi token-token dan hitung berapa jumlah token yang ada.

**Input.**

**Input.**

Kalimat : Nama saya Maharani Putri Efendi

**Output :**

Jumlah Token : 5 yaitu:

Nama

Saya

Maharani

Putri

Efendi

5. Masukkan kalimat dan pecah kalimat menjadi token-token. Cari yang sesuai dengan kata depan dan balik sisa string.

**Input.**

Kalimat : kancil kartu kamu kantor kisah kantin kita

Kata depan : kan

**Output.**

kanlic  
kanor  
kannit

6. Masukkan kalimat dan pecahlah kalimat menjadi token-token terlebih dahulu, selanjutnya reverse token-token tersebut dan tambahkan diakhir token dengan kata diminta.

**Input.**

Kalimat : warna balonku merah kuning biru hijau  
Kata : ay

**Output :**

Jumlah Token : 6 yaitu:  
anraway  
uknolabay  
haremay  
gninukay  
uribay  
uajihay

7. Masukkan sebuah kalimat. Ubahlah sebuah pola tertentu pada kalimat tersebut dengan pola yang lain.

**Input.**

Kalimat : warna darsa marka kaliarsa arti marwah  
Pola asal : ar  
Diganti dengan pola : satu

**Output.**

Kalimat : *wsatuna dsatusa msatuka kalisatusa satuti msatuwah*

8. Masukkan sebuah kalimat. Hapuslah pola tertentu pada kalimat tersebut berdasarkan pola yang diinputkan.

**Input.**

Kalimat : warna darsa marka kaliarsa arti marwah  
Pola yang dihapus : ar

**Output.**

Kalimat : *wna dsa mka kalisa ti mwah*

9. Masukkan dua kata. Pecah sebuah kata menjadi dua bagian, pecahlah secara random begitu juga dengan kata2.

Input kata1 = kata11 + kata 12  
kata2 = kata21 + kata 22

**Output**

Output1 = kata 11 + kata22  
Output2 = kata 12 + kata 21

**Input.**

Kata1 : abcdefghijkl

Kata2 : ghijklmnopqrstuvw

**Output.**

Output1 : abcde-opqrstuvw

Output2 : fghijkl-ghijklmn

10. Terdapat sebuah kunci dan sebuah kata/kalimat, dengan kunci tersebut maka kalimat akan menjadi kata sandi.

**Kunci :**

abcdefghijklmnopqrstvwxyz

n=2 yzabcdefghijklmnopqrstvw

n=7 uvwxyzabcdefghijklmnopqrst

n=20 ghijklmnopqrstvwxyzabcdef

**Kalimat :** Belajar OOP

**N=7**

**Output :**

**Kalimat :** vyfudul iij

11. Palindrom adalah sebuah kata, frasa, angka maupun susunan lainnya yang dapat dibaca dengan sama baik dari depan maupun belakang (spasi antara huruf-huruf biasanya diperbolehkan). Kata "palindrom" berasal dari bahasa Yunani: palin ("balik") dan dromos ("pacuan kuda").

**Contoh :**

**Kalimat yang termasuk palindrom misalnya:**

Kasur ini rusak.

Kasur Nababan rusak.

**Kata-kata yang termasuk palindrom misalnya:**

ada, apa, ara, asa, bab, ini, katak, kodok, makam, malam, radar, taat, dll

## Soal String Lanjut

1. Created by : PCLP

Alphabetic lovers found some random unordered alphabet. Please help them to arrange that alphabetic string to ordered alphabet. Lowercase placed in front of (left) uppercase if the letter same.

### Input

The first line of input specifies the number  $N$  of ranges, and each of the  $N$  following lines contains unordered alphabetic string which is no more than 100 characters length for each string.

### Output

For each alphabetic string, print the ordered string.

### Sample Input

```
5
hrAvaO
AaBbCcDd
ABC
CAB
A
```

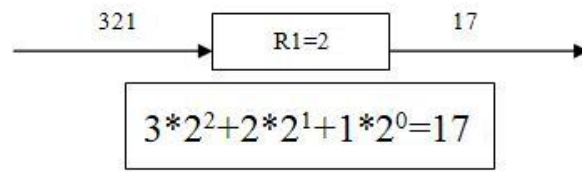
### Sample Output

```
aAhOrv
aAbBcCdD
ABC
ABC
ABC
A
```

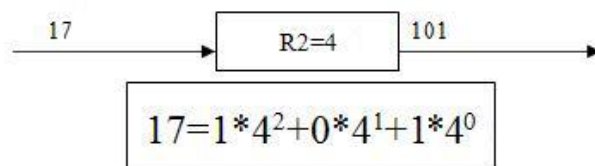
2. Converter (Created by : PCLP )

**ITS Lab** produced a batch of number system converter recently. There are two kind of converter R1 and R2.

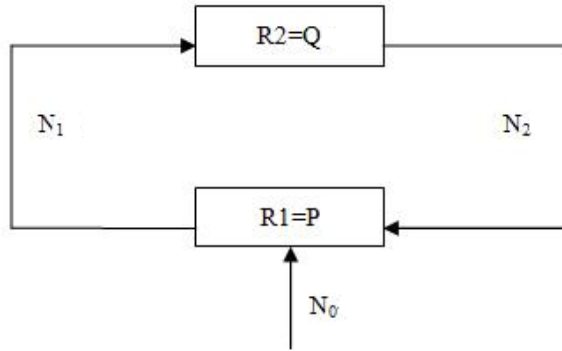
Given a parameter of  $P$ , converter R1's function is to transform the incoming  $P$  based integer into decimal number system. To simplify the design, R1 will not check that if the input is legal in  $P$  number system.



R2 is similar to R1, while acting an inverse function: convert a decimal number to  $Q$  (specified by R2) number system.



Mr. R, a researcher of ITS Lab, made a circuit uses R1 and R2:



(The circuit works at an order R1, R2, R1, R2 ...)

And he found an interesting phenomenon: if  $P < Q$  then for any input sequence  $N_0$ , after plenty of convention  $N_1$  will finally equal to  $N_2$ . For example,  $P=2$ ,  $Q=4$ ,  $N_0=321$ , finally,  $N_1=N_2=3$ .

Mr. R is excited for his discover, and wants to know what  $N_1$  and  $N_2$  will finally be in this circuit when  $P$ ,  $Q$  and  $N_0$  is given.

**Input**

The input contains several test cases. The first line is the number of test cases. The first line of each test case specifies two number  $P$  and  $Q$  ( $1 < P < Q < 37$ ), follow by  $N_0$  (the length of  $N_0$  is not larger than 5,000,000). Note that a digit larger than 9 is given in small letters, e.g. 'a'=10, 'b'=11 ... and so on.

**Output**

For each test case, output a line standing for the final sequence in the circuit.

**Sample Input**

```
2
2 4
321
19 25
3888175
```

**Sample Output**

```
3 m
```

\*\*\*\*\* Selamat Mengerjakan \*\*\*\*\*