

Soal Praktikum Bab IV

1. Buatlah sebuah fungsi yang menerima sebuah kata sebagai input dan menghasilkan deret permutasi dari kata tersebut. Setiap langkah dalam deret permutasi menggeser karakter pertama ke posisi terakhir. Tampilkan semua langkah permutasi dalam satu baris dan pisahkan dengan tanda '|'. Aplikasikan exception handling untuk mencegah error dalam kasus input yang tidak valid (TypeError).

Test Case 1:

```
stringPermutation("Mobil")  
lMobi | ilMob | bilMo | obilM | Mobil |
```

Test Case 2:

```
stringPermutation("Ayam")  
mAya | amAy | yamA | Ayam |
```

2. Buatlah sebuah fungsi Python yang dapat memeriksa apakah sebuah kata adalah palindrom atau bukan. Fungsi ini akan menerima satu parameter berupa string yang merupakan kata yang ingin diperiksa. Fungsi akan mengembalikan string "Palindrom" jika kata tersebut adalah palindrom, dan "Bukan Palindrom" jika bukan. Aplikasikan type hinting dalam pembuatan fungsinya.

Test Case 1:

```
palindrom("Radar")  
Palindrom
```

Test Case 2:

```
palindrom("Step on no pets")  
Palindrom
```

3. Buatlah sebuah fungsi Python yang dapat menemukan angka terbesar dalam sebuah daftar angka. Namun, Anda tidak diperkenankan menggunakan fungsi bawaan Python seperti `max()` atau `sorted()`. Fungsi ini harus menerima daftar angka sebagai argumen dan mengembalikan angka terbesar dalam daftar tersebut sebagai hasilnya.

Test Case 1:

```
maksimum(1,2,4,6,9,3,1,9,10)  
>> 10
```

Test Case 2:

```
maksimum(0,1,90,430,23,212,34)  
>> 430
```

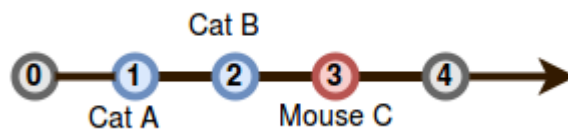
4. Dalam soal ini, ada dua kucing dan seekor tikus yang berada pada posisi-posisi berbeda di garis. Anda akan diberikan posisi awal kucing dan tikus. Tugas Anda

adalah membuat fungsi `catAndMouse` untuk mengembalikan jawaban yang sesuai untuk setiap permintaan :

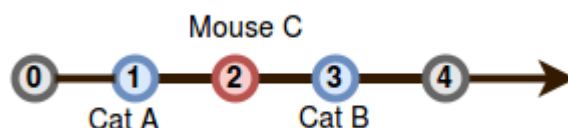
- Jika kucing A yang lebih dulu menangkap tikus, cetak "Cat A".
- Jika kucing B yang lebih dulu menangkap tikus, cetak "Cat B".
- Jika kedua kucing mencapai tikus pada saat yang sama, cetak "Mouse C" karena kedua kucing akan bertarung dan tikus akan lolos.

Contoh:

- Pada contoh pertama, kucing A berada pada posisi 1, kucing B berada pada posisi 2, dan tikus berada pada posisi 3. Kucing B akan lebih dulu mencapai tikus karena jaraknya hanya 1 unit, sementara kucing A harus menempuh jarak 2 unit. Oleh karena itu, kita mengembalikan "Cat B".



- Pada contoh kedua, kucing A berada pada posisi 1, kucing B berada pada posisi 3, dan tikus berada pada posisi 2. Kedua kucing akan mencapai tikus pada saat yang sama, tetapi karena tikus bisa lolos, kita mengembalikan "Mouse C".



Test Case 1:

```
catAndMouse(catA = 16, catB = 24, mosC = 15)
```

```
Cat A
```

Test Case 2:

```
catAndMouse(catA = 20, catB = 10, mosC = 15)
```

```
Mouse C
```