

## 1. 피보나치 함수

다음 소스는 N번째 피보나치 수를 구하는 파이썬 함수이다.

```
def fibonacci(n):  
    if(n==0):  
        return 0  
    elif(n==1):  
        return 1  
    else:  
        return fibonacci(n-1) + fibonacci(n-2)
```

fibonacci(3)을 호출하면 다음과 같은 일이 일어난다.

- fibonacci(3)은 fibonacci(2)와 fibonacci(1) (첫 번째 호출)을 호출한다.
- fibonacci(2)는 fibonacci(1) (두 번째 호출)과 fibonacci(0)을 호출한다.
- 두 번째 호출한 fibonacci(1)은 1을 출력하고 1을 리턴한다.
- fibonacci(0)은 0을 출력하고, 0을 리턴한다.
- fibonacci(2)는 fibonacci(1)과 fibonacci(0)의 결과를 얻고, 1을 리턴한다.
- 첫 번째 호출한 fibonacci(1)은 1을 출력하고, 1을 리턴한다.
- fibonacci(3)은 fibonacci(2)와 fibonacci(1)의 결과를 얻고, 2를 리턴한다.

1은 2번 출력되고, 0은 1번 출력된다. N이 주어졌을 때, fibonacci(N)을 호출했을 때, 0과 1이 각각 몇 번 출력되는지 구하는 프로그램을 작성하시오.

[입력]

첫째줄에 테스트 케이스의 개수 T가 주어진다.

각 테스트 케이스를 한 줄로 이루어져 있고, N이 주어진다.

[출력]

각 테스트 케이스마다 0이 출력되는 횟수와 1이 출력되는 횟수를 공백으로 구분해서 출력한다.

```
3 <- T  
0 <- N  
0 0  
1 <- N  
0 1  
3 <- N  
1 2
```

## 2. 분산처리

10대의 컴퓨터가 있다. 각 컴퓨터에 1번부터 10번까지의 번호를 부여하고, 10대의 컴퓨터가 다음과 같은 방법으로 데이터들을 처리하기로 하였다.

1번 데이터는 1번 컴퓨터, 2번 데이터는 2번 컴퓨터, 3번 데이터는 3번 컴퓨터, ... ,

10번 데이터는 10번 컴퓨터, 11번 데이터는 1번 컴퓨터, 12번 데이터는 2번 컴퓨터, ....

총 데이터의 개수는 항상  $a \times b$ 개의 형태로 주어진다. 마지막 데이터가 처리될 컴퓨터의 번호를 구하시오.

[입력]

정수  $a$ 와  $b$ 가 주어진다.

[출력]

각 테스트 케이스에 대해 마지막 데이터가 처리되는 컴퓨터의 번호를 출력한다.

[예제 입력]

A	B	출력
1	6	1
3	7	7
6	2	6
7	100	1
9	635	9

### 3. 다이얼



다음과 같은 오래된 다이얼 전화기가 있다.

어떠한 단어가 입력될 때, 각 알파벳에 맞는 숫자의 합이 출력되어진다.

예를 들어, UNUCIC가 입력될 경우  $8+6+8+2+4+2 = 30$ 이 출력된다.

각 단어에 맞는 숫자의 합을 구하시오.

입력	출력
UNUCIC	30
APPLE	24
PYTHON	40

#### 4. 버블정렬

버블 정렬은 서로 인접해 있는 두 수를 바꿔가며 정렬하는 방법이다. 예를 들어 [3,2,1]이 있을 때 버블 정렬의 과정은 다음과 같다

[3,2,1]

[2,3,1]

[2,1,3]

[1,2,3]

다음의 빈곳을 채우시오.

```
def bsort(s):  
    //fill here!  
  
print(bsort([32,23,18,7,11,99,55]))  
print(bsort([12,4,6,1,31,223,16,17]))  
print(bsort([142,236,213,27,18,15]))  
print(bsort([17,5,1,3,21,45,26,7]))  
print(bsort([71,532,31,23,6,2,37,13]))
```

[출력값]

```
[7, 11, 18, 23, 32, 55, 99]  
[1, 4, 6, 12, 16, 17, 31, 223]  
[15, 18, 27, 142, 213, 236]  
[1, 3, 5, 7, 17, 21, 26, 45]  
[2, 6, 13, 23, 31, 37, 71, 532]
```