

2021년 1월 10일 일요일 오후 6:54

첫째 줄에 테스트 케이스의 개수 t ($1 \leq t \leq 100$)이 주어진다. 각 테스트 케이스는 한 줄로 이루어져 있다. 각 테스트 케이스는 수의 개수 n ($1 < n \leq 100$)가 주어지고, 다음에는 n 개의 수가 주어진다. 입력으로 주어지는 수는 1,000,000을 넘지 않는다.

각 테스트 케이스마다 가능한 모든 쌍의 GCD의 합을 출력한다.

예제 출력 1 복사

소스코드

최대공약수 공식 : 유클리드 호제법

```
def GCD(a, b):
```

```
if a<b:
```

$$a, b = b, a$$

```
while b!=0:
```

$$a, b = b, a \% b$$

```
return a
```

1. 반복할 값 입력

```
cycle = int(input())
```

2. cycle만큼 숫자를 입력받는다

```
cycle_arr = []
```

```
for i in range(cycle):
```

```
cycle_arr.append([int(x) for x in input("").strip().split()])
```

3. 입력받은 숫자로 최대공약수를 구해 합을 출력

```
#print(cycle_arr)
```

```
for cycle in cycle_arr:
```

```
sum = 0
```

```
my_cycle = cycle[0]-1
```

```
for i in range(my_cycle):
```

```
for j in range(my_cycle-i):
```

```
sum += GCD(cycle[i+1], cycle[j-i+2]);
```

```
print(sum)
```

j 는 0 시작
나머지 기종수, +1하여
뒤 수와 GCD