

happy unhappy

구하고자 하는 것 : 주어진 인풋 분류

input $n \rightarrow f(n)$ $\begin{cases} \text{happy} \\ \text{unhappy} \end{cases}$

$$f(n) = \begin{cases} 1 & \text{if } f(n) == 1 \\ f(f(n)) & \text{if } f(n) > 1 \\ 0 & \text{if } f(n) = 0 \end{cases}$$

recursive + Dict

Loop t--
cal(input)
Cal:
1 -> ret tr
dict -> cont
return fa

아주 간단한 리커시브 문제였다. 반복이 되는지 확인하는 방법은 딕셔너리(hash map)을 통하여 빠르게 검색할 수 있게 하였다.

python dict의 contains()를 사용 하였다. 다른 언어들의 맵은 어떻게 사용하는지 알아보고 복습하자.

```
def cal(n,d):
```

```
    if n == 1 : return True
```

```
    if d.__contains__(n): return False
```

```
    t = n
```

```
    nextN = 0
```

```
    while t > 0 :
```

```
        nextN += int(pow((t%10),2))
```

```
        t //= 10
```

```
    d.setdefault(n,1)
```

```
    return cal(nextN,d)
```

```
if __name__ == "__main__":
```

```
    t = int(input())
```

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

```
d = {}  
n = int(input())  
if cal(n,d) : print("HAPPY")  
else : print("UNHAPPY")
```