

블랙잭 문제

카지노에서 자주 플레이되는 블랙잭.

카드의 합이 21이 넘지 않는 한도 내에서, 카드의 합을 가장 크게 만드는 게임

- 첫째 줄에는 카드의 개수 N 과 목표 값 X 가 주어지며, 두번째 줄에는 카드에 쓰여 있는 수가 입력으로 주어질 때, X 에 가장 가까우면서 X 를 넘지 않는 카드 세 장의 합을 구해 출력하세요.

사용되는 파이썬 함수들

- `split()` 함수: 문자열을 분리할 때 사용, 구별자를 지정하지 않으면 공백문자를 기준으로 문자열을 분리하여 리스트로 반환
- `map()` 함수: 첫 번째 인자로 함수의 이름, 두 번째 인자로 여러 개 요소가 담긴 일련의 자료를 전달하여 여러개의 요소를 하나씩 개별적으로 함수 적용하지 않아도 되는 편리한 기능
저장할 변수 = `map(함수 이름, 입력 집합인 시퀀스 자료형)`
저장할 리스트 = `list(map(함수 이름, 리스트))`
- `append()` 함수: 넘겨받은 인수의 자료형에 상관없이 리스트 뒤에 그대로 추가
- `range(시작숫자, 끝숫자)`: 숫자 리스트를 자동으로 만들어주는 함수

```

n, x = map(int, input('카드의 개수와 목표 값: ').split())
num = list(map(int, input('카드에 쓰여있는 수: ').split()))
sum_list = []
sum = 0

for i in range(n):
    for j in range(i+1, n):
        for k in range(j+1, n):
            sum = num[i] + num[j] + num[k]
            if sum < x:
                sum_list.append(sum)
print(max(sum_list))

```

- 3중 for문을 사용하여 모든 경우의 수를 검사하고, 세 카드의 합이 x보다 작으면 리스트에 요소 추가, 모든 경우의 수를 검사 후 리스트에 저장되어 있는 값 중 가장 큰 값을 출력

```

===== RESTART: C:/Users/yeram/OneDrive/바탕 화면/gg.py =====
====
카드의 개수와 목표 값: 5 21
카드에 쓰여있는 수: 2 3 4 5 6
15
>>>
===== RESTART: C:/Users/yeram/OneDrive/바탕 화면/gg.py =====
====
카드의 개수와 목표 값: 7 50
카드에 쓰여있는 수: 5 10 15 20 25 30 35
45

```