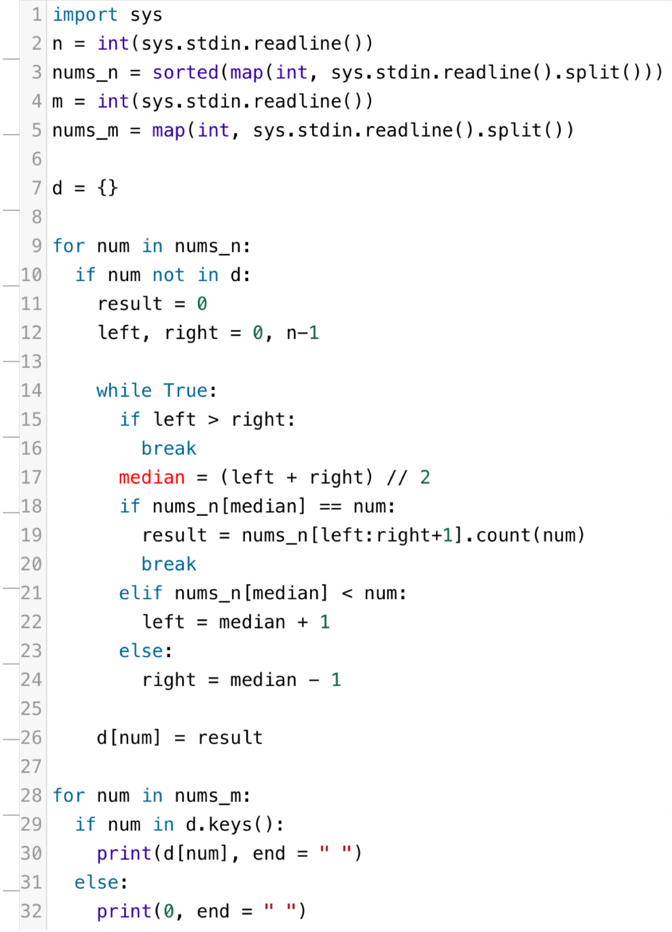
〈 이분탐색 〉

1. 숫자카드 2



```python
import sys
n = int(sys.stdin.readline())
nums_n = sorted(map(int, sys.stdin.readline().split()))
m = int(sys.stdin.readline())
nums_m = map(int, sys.stdin.readline().split())

d = {}

for num in nums_n:
  if num not in d:
    result = 0
    left, right = 0, n-1

    while True:
      if left > right:
        break
      median = (left + right) // 2
      if nums_n[median] == num:
        result = nums_n[left:right+1].count(num)
        break
      elif nums_n[median] < num:
        left = median + 1
      else:
        right = median - 1

    d[num] = result

for num in nums_m:
  if num in d.keys():
    print(d[num], end = " ")
  else:
    print(0, end = " ")
```

nums_n = [-10, -10, 2, 3, 3, 6, 7, 10, 10, 10]

num = 3　　nums_m [10, 9, -5 · · ·]

→ left = 0. right = 9 ⇒ median : 4

nums_n[4] = 3 == 3

∴ [-10, -10, 2, 3, 3, 6, 7, 10, 10, 10].count[3] → 2

d = {3 : 2}

num = 10

→ left = 0. right = 9 ⇒ median : 4

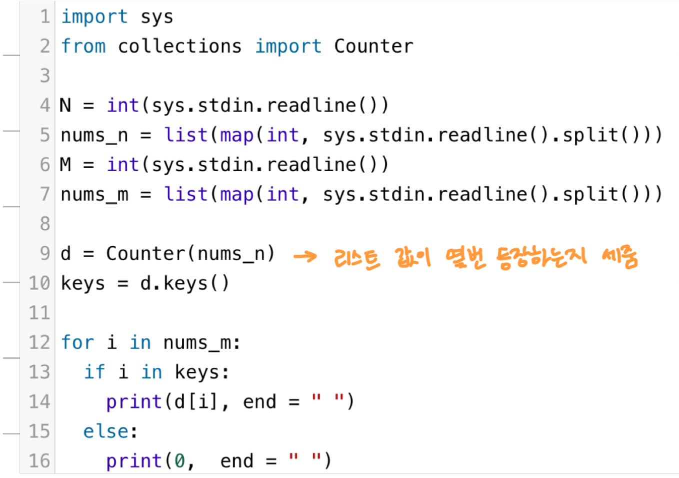
nums_n[4] = 3 < 10

→ left = 5. right = 9 ⇒ median : 7

nums_n[7] = 10 == 10

∴ [6, 7, 10, 10, 10].count[10] → 3

* 아주 간단한 방법



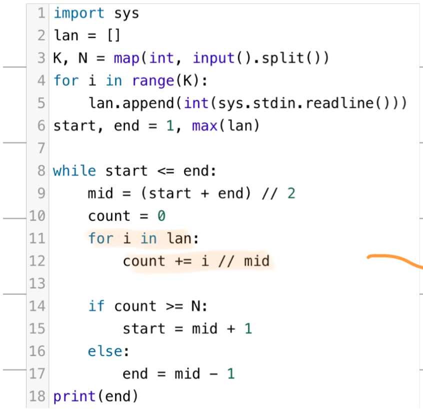
```python
import sys
from collections import Counter

N = int(sys.stdin.readline())
nums_n = list(map(int, sys.stdin.readline().split()))
M = int(sys.stdin.readline())
nums_m = list(map(int, sys.stdin.readline().split()))

d = Counter(nums_n)
keys = d.keys()

for i in nums_m:
  if i in keys:
    print(d[i], end = " ")
  else:
    print(0,  end = " ")
```

→ 리스트 값이 몇번 등장하는지 세줌

nums_n = [-10, -10, 2, 3, 3, 6, 7, 10, 10, 10]

d = { -10:2 , 2:1, 3:2, 6:1, 7:1, 10:3 }

2. 랜선 자르기



```python
import sys
lan = []
K, N = map(int, input().split())
for i in range(K):
    lan.append(int(sys.stdin.readline()))
start, end = 1, max(lan)

while start <= end:
    mid = (start + end) // 2
    count = 0
    for i in lan:
        count += i // mid

    if count >= N:
        start = mid + 1
    else:
        end = mid - 1
print(end)
```

K = 4, N = 11

lan = [802, 743, 457, 539]

(start = 1, end = 802, mid = 401) →

→ mid 길이로 잘랐을 때 나오는 랜선 개수 : count

802 → 2개, 743 → 2+1개, 457 → 2+1+1개, 539 → 2+1+1+1 개

∴ 총 5개 < 11 → end = 401 − 1 = 400

(더 많은 랜선이 나와야 하므로 더 짧은 길이로 잘라야 함)

start = 1 end = 400 mid = 200

총 11개

start = 200+1 = 201

start = 201 end = 400, mid = 300 6 < 11 ∴ end = 299

start = 201 end = 299, mid = 250 8 < 11 ∴ end = 249

start = 201 end = 249, mid = 225 10 < 11 ∴ end = 224

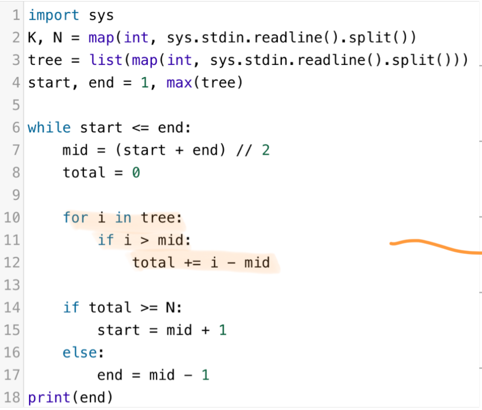
start = 201 end = 224, mid = 212 10 < 11 ∴ end = 211

start = 201 end = 211, mid = 206 10 < 11 ∴ end = 205

start = 201 end = 205, mid = 203 10 < 11 ∴ end = 202

start = 201 end = 202, mid = 201 10 < 11 ∴ end = 200

3. 나무자르기



```python
import sys
K, N = map(int, sys.stdin.readline().split())
tree = list(map(int, sys.stdin.readline().split()))
start, end = 1, max(tree)

while start <= end:
    mid = (start + end) // 2
    total = 0

    for i in tree:
        if i > mid:
            total += i - mid

    if total >= N:
        start = mid + 1
    else:
        end = mid - 1
print(end)
```

N = 4, K = 7

tree = [20, 15, 10, 17]

start = 1, end = 20, mid = 10

→ mid 길이로 잘랐을 때 나오는 나무조각들의 총 길이 : total

20 → 10, 15 → 10+5, 10 pass 17 → 10+5+7

총 22 >= 7 ⇒ start = 10+1

(더 긴 길이가 나왔으므로 절단기 높이를 올려도 됨)

start = 11, end = 20, mid = 15 7 == 7 start = 16

start = 16, end = 20, mid = 18 2 < 7 end = 17

start = 16, end = 17, mid = 16 5 < 7 end = 15