




[BOJ] 1253. 좋다

 Date	@January 20, 2025
 Problem Link	https://www.acmicpc.net/problem/1253
 Assignment Link	https://www.acmicpc.net/problem/18114

문제

1253.좋다

N 개의 수 중에서 어떤 수가 **다른 수 두개의 합**으로 나타낼 수 있다면 그 수를 "좋다(GOOD)"고 한다.

N 개의 수가 주어지면 그 중에서 좋은 수의 개수는 몇 개인지 출력하라.

수의 위치가 다르면 값이 같아도 다른 수이다.

입력

- 수의 개수 N ($1 \leq N \leq 2,000$)
- i 번째 수를 나타내는 A_i N 개 ($|A_i| \leq 1,000,000,000$, A_i 는 정수)

출력

좋은 수의 개수

예제

```
# 예제 입력
10
1 2 3 4 5 6 7 8 9 10

# 예제 출력
8
```

→ 3, 4, 5, 6, 7, 8, 9, 10이 "좋다"

▼ 이유

- $3 = 1 + 2$
- $4 = 1 + 3$
- $5 = 2 + 3$
- $6 = 1 + 5$ 또는 $2 + 4$
- $7 = 3 + 4$
- $8 = 3 + 5$ 또는 $2 + 6$
- $9 = 4 + 5$
- $10 = 4 + 6$

아이디어



문제 유형

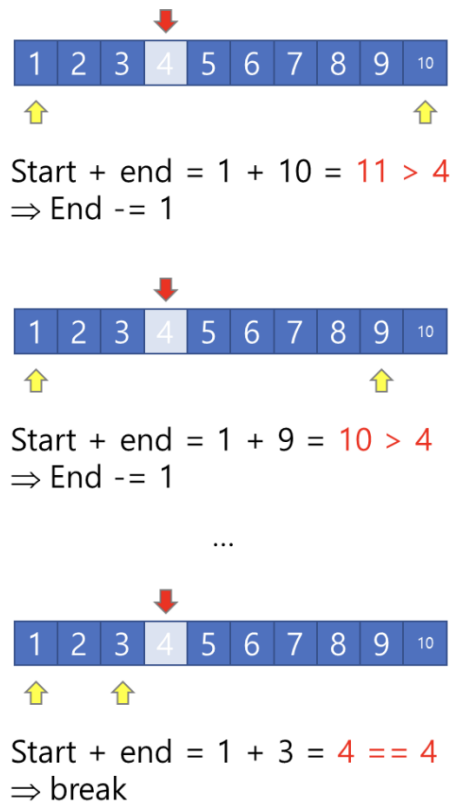
투 포인터(Two Pointer)

: 두 원소의 합이 선택한 원소와 같은지 비교한다.



아이디어

$$A[k] = A[i] + A[j] \quad (i \neq k, j \neq k)$$



주의

- 배열을 오름차순 정렬한다. 투 포인터 알고리즘은 정렬된 배열에서 합을 계산하며 탐색할 때 효율적으로 동작한다.
- 정렬된 데이터에서 **자기 자신**을 좋은 수 만들기에서 제외해야한다.
투 포인터의 위치가 현재 확인 중인 수와 같다면 건너뛴다.

정답

- 각 숫자 `goal` 에 대해 `start` 와 `end` 를 초기화하고, `goal` 을 만족하는 두 수의 합을 탐색한다.
- `start` 와 `end` 가 `goal` 을 가리키지 않도록 처리한다.
- "좋은 수" 조건을 만족하면 `cnt` 를 증가시킨다.

풀이

```
import sys
input = sys.stdin.readline

N = int(input()) # 수의 개수
num = list(map(int, input().split())) # 입력된 숫자 리스트
```

```

num.sort() # 정렬 (투포인터 사용을 위해)

cnt = 0 # 좋은 수의 개수
for i in range(N):
    goal = num[i] # 현재 확인할 값
    start = 0
    end = N - 1


    while start < end: # 두 포인터가 교차할 때까지 반복
        # 현재 두 수의 합이 goal인지 확인
        if num[start] + num[end] == goal:
            if start == i: # start 포인터가 현재 값(goal)을 가리키면 이동
                start += 1
            elif end == i: # end 포인터가 현재 값(goal)을 가리키면 이동
                end -= 1
            else: # 두 수의 합이 goal이면서 현재 값(goal)을 포함하지 않을 때
                cnt += 1
                break

        elif num[start] + num[end] > goal: # 합이 goal보다 크면 큰 값을 줄여야 하므로 end 감소
            end -= 1

        else: # 합이 goal보다 작으면 작은 값을 키워야 하므로 start 증가
            start += 1

print(cnt)

```

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이	제출한 시간
88913122	learntosurf	 1253	맞았습니다!!	32412 KB	1276 ms	Python 3 / 수정	1128 B	11초 전

- 투 포인터 초기화
 - `start`: 배열의 시작점 (0부터 시작)
 - `end`: 배열의 끝점 ($N-1$ 부터 시작)
 - `goal`: 현재 좋은 수인지 확인할 목표 값 (`num[i]`)
- 투 포인터 탐색
 - 조건1) 두 수의 합이 `goal` 과 같으면

`goal` 자신을 포함하지 않도록 확인하고 "좋은 수"로 카운트한다.

`cnt` 를 증가시키고 루프를 종료한다.
 - 조건2) 두 수의 합이 `goal` 보다 크면 더 작은 합을 만들기 위해 `end` 를 감소시킨다.
 - 조건3) 두 수의 합이 `goal` 보다 작으면 더 큰 합을 만들기 위해 `start` 를 증가시킨다.
- 모든 `num[i]`에 대해 위 과정을 반복한 뒤, `cnt` 값을 출력한다.

과제

18114. 블랙 프라이데이