투포인터

Created	@October 27, 2021 6:48 PM
: ≡ Tags	투포인터
≡ 난이도	골드4

투포인터

투포인터는 반복문 안에서 두 개의 포인터를 독립적으로 이동시키는 알고리즘이다.

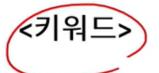
각 포인터들에게 의미를 부여하는게 중요하다.

예를들어서

```
int N = 5;
int l = 1, r = N;
while (true) {
  if (l + r > N) l++;
  else if (l + r < N) r++;
  else break;
}</pre>
```

이런식이다.

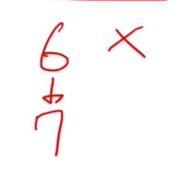
문제를 읽다가 연속 부분 수열 Or 순서를 지키며 차례대로 , 곱의 최소 라는 키워드가 나오면 투 포인터를 고려해보자.



- 1차원 배열에서의 "연속 부분 수열" or "순서를 지키며 차례대로"
- 곱<u>의 최소</u>







=> 이런 단어가 등장하면 Two Pointers 접근을 시도해 볼 가치가 있다!

[BOJ] <u>부분합 1806</u>

투포인터 1





시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
0.5 초 (하단 참고)	128 MB	38655	10155	7163	25.318%

☆ 한국어 ▼

문제

10,000 이하의 자연수로 이루어진 길이 N짜리 수열이 주어진다. 이 수열에서 연속된 수들의 부분합 중에 그 합이 S 이상이 되는 것 중, 가장 짧은 것의 길이를 구하는 프로그램을 작성하시오.

연속 부분 수열 , 최소 키워드 등장

풀이 1. 완전탐색

수열의 길이는 최대 100_000 모두 탐색하려면 O(10_000!) 연산 1억번 당 1초니까 0.5초 안에 패스 못함

풀이 2. 투포인터

왼쪽 포인터는 수열의 왼쪽 끝을 나타낸다. 오른쪽 포인터는 수열의 오른쪽 끝을 나타낸다. 추가로 현재까지 수열의 합을 보관하는 변수를 정의한다.

코드보기.

예제 입력의 함정

예제 입력 1 복사

```
10 15
5 1 3 5 10 7 4 9 2 8
```

M = 15니까 왼쪽 포인터가 (두번 째)5, 오른쪽 포인터가 10일 때 정답이다. 그런데 만약 10이 15로 바뀐다면?

```
▼ 시뮬레이션
```

R

L

5 1 3 5 15 7 4 9 2 8

```
L R
5 1 3 5 15 7 4 9 2 8
L R
5 1 3 5 15 7 4 9 2 8
      R
5 1 3 5 15 7 4 9 2 8
                    14
L R
                     28
5 1 3 5 15 7 4 9 2 8
        R
                     23
5 1 3 5 15 7 4 9 2 8
    L R
5 1 3 5 15 7 4 9 2 8
                     22
     L R
                     20
5 1 3 5 15 7 4 9 2 8
        R
5 1 3 5 15 7 4 9 2 8
                     15
       L R
                    17
5 1 3 5 10 7 4 9 2 8
```

투포인터 꿀팁

- 1. 왼쪽 포인터의 증가를 for 문에게 위임하자
- 2. 왼쪽 포인터를 기준으로 생각하자

```
int L = 1, R = 1;
while (왼쪽 포인터 조건) {
  while (오른쪽 포인터 조건) {
  }
}
```

두개의 포인터를 조작해야 한다. 반면에 for 문을 사용하면?

```
int R = 1;
for (int L = 1; l <= N; l++ {
 while (오른쪽 포인터 조건) { }
}
```

생각해야 하는 범위가 확 줄어든다.

지금까지 푼 문제들은 왼쪽 포인터를 기준으로 하는게 편했다. 일단 왼쪽 포인터는 그냥 전진한다고 생각하자. 왼쪽 포인터를 기준으로 하지 않으니까 실수도 많이 발생함

투포인터

3