0821

**1) 프로그래머스 문자열 압축 문제**

배운점)

* Index를 정확하게 파악하지 않고 테스트해서 경험적으로 고치는 습관!
* 문제를 정확하게 읽지 않고 푼다. 인덱스를 하나씩 추가해야할 경우는 없었는데, 있는 것으로 착각하여 예외처리를 했다.

잘한점)

* 이전에 풀었던 코드보다 한층 가독성이 좋아졌다! 아이디어를 쉽게 떠올렸다. Ex) 반복문을 통해서 1 단위로 압축, 2 단위로 압축, 3 단위로 압축하면 되겠구나.

**2) 프로그래머스 무지의 먹방 라이브**

배운점)

* 우선순위 큐를 써야하는 아이디어를 떠올리지 못했다. -> 작은 값을 배열에서 선택하고, 삭제도 해야한다면 우선순위 큐가 유리함. (딕셔너리로 구현해봤는데 매우 어려웠음. )
* 무조건 구현으로만 해결할 수 있다고 생각했다. 이 문제에서는 그리디 접근으로 푸는 아이디어 문제!

**3) 백준 피보나치 수열**

배운점)

* 재귀에 대한 이해 + 연산한 적이 있다면 바로 return한다. (다이나믹 아이디어)

**4) 프로그래머스 입국 심사 문제**

배운점)

* 역으로 나누기 연산 사용하기 -> 전체 시간에 배열의 요소들을 나누어 합하면 몇 명을 심사할 수 있는지가 나온다.
* Start, mid는 최소 최대가 있고 왔다 갔다 하는 흐름으로 결정할 수 있는 문제여야한다.
* 최소를 구하는 문제라면 (정확히 정해진 값이 아닌) mid = n 대신에 >=n 으로 로직을 짜면 최적의 해를 구할 수 있다.

0822

**1) 의상**

배운점)

* 딕셔너리 사용법 : 딕셔너리 value 안에 list를 넣을 수 있다. (append도 가능)

**2) 공유기 설치**

배운점)

* 아이디어는 맞았으나 구현 (어떤 인덱스를 return 할지 어려움.)이 부족

**3) 디스크 컨트롤러**

잘한점)

* 생각 그대로 구현했더니 풀렸다! 큐에 넣고 빼고!

**4) 체육복**

배운점)

* 쉬운 문제었는데 예외 케이스를 못찾음.
* 중복을 체크할 때는 set 연산을 사용해도 좋다!