

# 10월 1주차 스터디

일정 : 10/4 ~ 10/10

멤버 : 오서하 김유진 우정연 오윤기

## 문제

- 복습 문제 : 개별적으로 다시 풀어보고 싶은 문제 2개 선택
  - 기출 문제 : 기출문제 2개 중 최소 1개 선택
1. 백준 마법사 상어와 블리자드 골드2 - <https://www.acmicpc.net/problem/21611>
  2. 프로그래머스 가장 긴 팰린드롬 레벨3 - <https://programmers.co.kr/learn/courses/30/lessons/12904>

## 풀이

### 김유진

#### B21611\_마법사상어와블리자드.java

- bead[] 배열로 시뮬레이션에 따른 구슬을 상태를 업데이트 해놓고 map[] 2차원 배열을 마법 사용하는 구간에서 break 조건으로 사용해서 한참 헤맸다.  
풀이법
1. map으로 bead 세팅
  2. 마법으로 구슬 파괴, 후 정렬
  3. 4개이상 붙어 있는 구슬들 파괴 및 점수 합산, 후 정렬
  4. 구슬 재정렬
  5. 2~4의 수행을 마법사용 횟수 m번 만큼 반복한다.
  6. 종료후 점수를 출력한다.

#### B2810\_컵홀더.java(복습)

- 출력에 컵을 컵홀더에 놓을 수 있는 최대 사람의 수를 출력한다. 라는 문구가 있음  
즉 사람들이 심보가 꼬여서 다른 사람이 컵홀더를 사용할 수 있는 상황을 막지는 않음  
그래서 간단하게 생각 가능함

#### B2531\_회전초밥.java(복습)

- 이전과 다르게 회전 행렬 부분은 %연산을 이용해서 구현함
- 이 문제에서는 선택한 k개 중에서 중복된 음식이 있음을 파악해야함
- 새로운 음식이 추가 되는 경우 sushi가 0->1  
기존에 선택한 음식이 완전히 사라지는 경우 1->0  
위의 두 가지 경우에 cnt(선택한 음식의 종류)를 업데이트 시키면 된다.

## 오서하

### B2531\_회전초밥.java(복습)

- 과제 때 틀림
- 큐를 이용해, 큐에 k개가 들어와 있다면 들어온 수는 visit체크를하면서 cnt를 세주었고, 해당 큐에 c가 없다면 추가적으로 cnt+1해주었음=> 시간초과

### 펠린드롬

- 문자열을 하나씩 증가시켜가며 팰린드롬여부를 체크함 글자수가 2일때는 0,1번 인덱스를 글자수 3일 때는 가운데 인덱스를 제외하고 맨끝매칭 맨끝-1끼리 여부 확인하여 팰린드롬을 확인함
- 이 과정까지는 맞았는데, 2중 for를 사용하여 모든 경우를 따져주어야 반례가 없다. 현재 코드에서는 abbaa인 경우, 가운데 b를 두고 (a,a)(b,a)를 매칭하기때문에 팰린드롬을 확인하지 못함=>시간초과가 걸릴것같아서 도전하지 않았고 다른 로직을 생각하려고 했음..(발표자 정연님이 완탐으로 성공하셨기에 도전해볼 예정)

## 우정연

### BJ1987\_알파벳(복습)

- dfs사용해 alpha boolean배열에 해당 알파벳을 지나갔는지의 여부를 저장

### BJ1992\_쿼드트리(복습)

- 분할정복문제 - 최종 결과 String을 반환해줌

### BJ21611\_마법사상어와블리자드

- 생각
  1. ArrayList / LinkedList쓰거나 배열에서 매번 배열값 당겨주면 시간초과 날 것 같다  
-> 다 없애거나 작업하고 마지막 한번씩 이동되도록
  2. ArrayList에 순서 저장 VS 배열에 저장(선택) VS 매번 순서 for문 돌리기
  3. 상어 탐색 방향과 입력방향이 다름 -> 입력방향 변환해주는 배열 changeDir
- 순서

1. Info 배열에 초기값들 추가 - 1 2번 2 2번 3 2번....
  2. 마법시전 - Info에서 해당 위치 찾아서 0으로
  3. 폭발 - Info배열에서 연속 4개인거 모두 0으로 만들 -> ans에 더함 + stack 사용
  4. 변화 - 새로운 배열 만들어서 여기에 추가
  5. 이동 - 매번 0이 된 칸을 당겨서 배열 다시배치
- 방법 : Method : stack, 반복문 하나로 인덱스 두개 사용해서 배열 2개 처리하기
  - 에러
    1. 문제 잘못읽음 - 연속 4개이상 없을까? 무한반복
    2. 4개 연속 아닐때도 stack 비워줘야 하는데 안함 - else stack.clear();
    3. (change)while문 안끝나는디..? -> info배열이 먼저 끝나는 경우를 고려 안해서 구슬 번호가 0인 경우도 끝나도록 처리
    4. (move)근데 새로 만들어서 옮기면 i, j 옮기는거랑 뒤에 null값 되는거랑 처리해야 함... -> 위치는 info 반복문에 맞춰서 저장하도록. 이후 구슬번호만 바꿔줌
    5. 짚어보니까 옮기다가 위치 꼬인듯. 메소드마다 info배열 짚어보면서 본문 결과와 비교 -> (move)idx를 nidx로 씀...
  - 느낀점
    - 코드 미리 안짜고, 변수랑 방법만 대충 생각해서 바로 코드로 옮겼더니 디버깅 좀 오래했다. 1시간..? 담부턴 더 구체적으로 계획하자...

### Pro연습\_가장긴팰린드롬

- 생각 - 문자열에서 모든 경우의 범위를 결정해 팰린드롬 여부를 확인. 해당 범위가 답보다 작으면 할 필요 X
- 방법 : 완전탐색
- 에러 : 효율성 테스트 2번째가 시간초과 남 -> 해당 범위가 답보다 작으면 검사할 필요 없게 함
- 느낀점 : 프로그래머스에서 작업하는게 안익숙한데 많이 해봐야겠다. 이 방법이 예제가 부족해서 통과하는 줄 알았는데 유진님 말씀 들어보니 괜찮은 방법인거 같다.

## 단독방 폭파

