2020년
ALPS 여름방학
비대면 스터디

- 7주차 -

문제풀이

참여방법

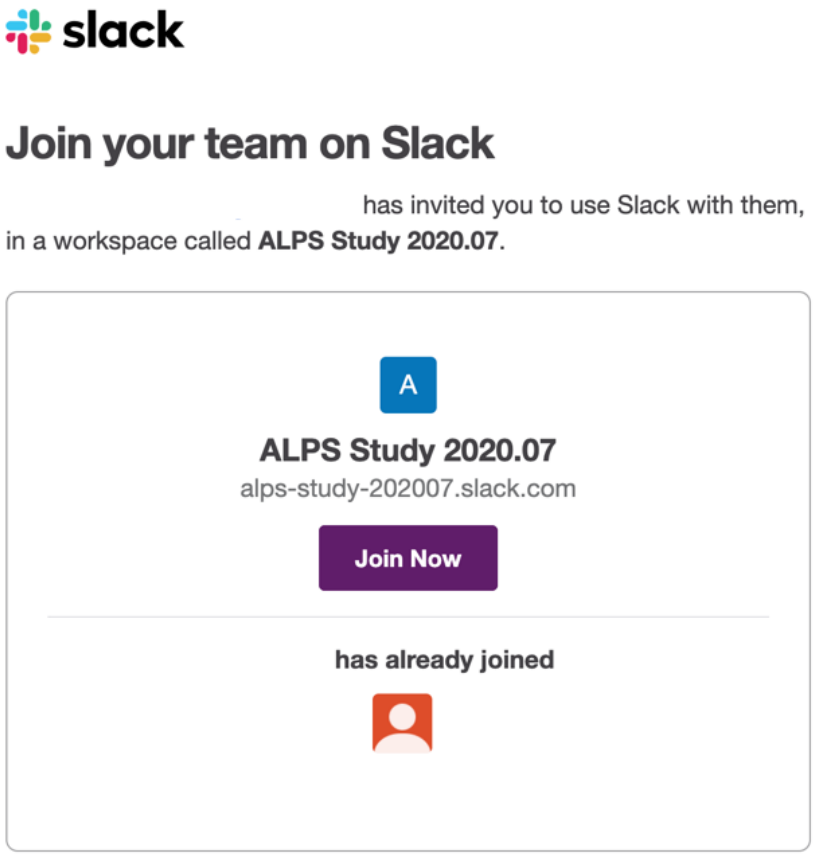
- 슬랙 - 코드 공유용으로 사용
- 구글 meet - 화면공유 및 음성으로 코드 설명 및 서로간 피드백

(https://meet.google.com/gpu-mixd-bqa) 링크는 계속 고정입니다.

항상 위 링크로 들어오시면 됩니다.

슬랙과 meet를 같이 띄우고 진행해주시면 감사하겠습니다.

참여방법 / 슬랙



메일로 보내드린
초대장에 Join Now를
클릭하시면
입장 가능합니다

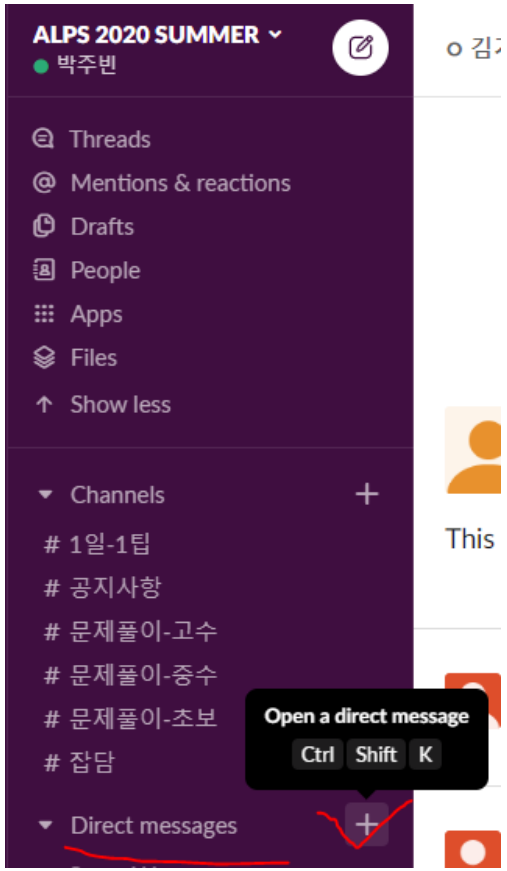
참여방법/구글 meet

- https://meet.google.com/gpu-mixd-bqa
- 해당 링크로 접속
- 해당 스터디 진행영상은 참여하지 못하신 분들을 위해 녹화 후 유튜브 채널에 업로드할 예정입니다^^
- 혹시 녹화를 원하지 않는 분들은 말씀해주세요.
- 아래는 영상이 올라갈 주소 입니다.
- https://www.youtube.com/playlist?list=PL9gVcwpebJSJJ80vNpdrYAhyG6PXxbgq_

시작하기 전에..

- 이번 7주차에는 먼저 **15분**간 개념 설명 뒤에
- **60분** 동안 모두에게 문제 푸는 시간이 주어집니다.
  - 도중에 궁금한 문제나 풀리지 않는 문제가 있다면 **슬랙방**에 질문을 하시고, **익명**으로도 가능하니 스터디 운영진에게 **Direct Message**를 통해 알려주세요.

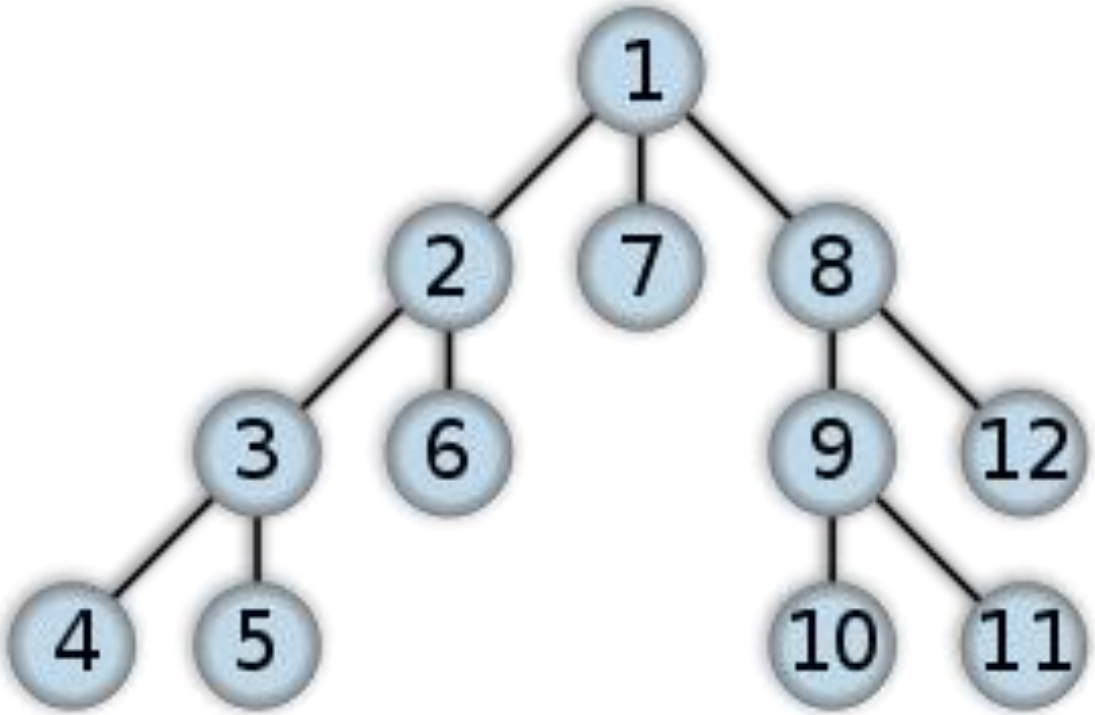
- 나머지 **40분**은 풀이 설명을 진행할 것입니다.
  - 시청은 **자유**이며, 여러분이 **해당 문제**의 설명을 듣지 않아도 된다고 판단하면 그 시간에 다른 문제를 푸셔도 좋습니다!



BFS

Breadth-First Search

너비 우선 탐색



앞 Queue 뒤

boj-1012 유기농 배추 (난이도 : 하)

https://www.acmicpc.net/problem/1012

boj-1012 유기농 배추

- 저번주 dfs문제를 bfs로 풀 수 있다.
- 상하좌우 오프셋만큼 각 배추밭에 대해서 bfs를 한다.

boj-1012 유기농 배추

- 큐가 빌 때까지 계속 돈다. 이때 새로운 칸들이 계속 큐에 들어온다.
- q.front()를 참조하고 q.pop()한 후 이 front()칸에 대해서 상하좌우 오프셋 중에 배추밭인 부분('1'인 부분)은 q.push()해준다.
- 이렇게 큐에 추가적으로 계속 칸들이 들어온다면 모든 배추밭을 탐색할 때까지 큐는 empty가 되지않는다.

boj-1012 유기농 배추

- https://github.com/alps-jbnu/ALPS_2020_Summer_Study/blob/master/KiwanKim/1012BFS.cpp

boj-17836 공주님을 구해라! (난이도 : 중상)

https://www.acmicpc.net/problem/17836

boj-17836 공주님을 구해라!

- 저번주 달이 차오른다 가자! 문제랑 비슷한 개념의 문제
- 저번주는 열쇠였다면 이번에 칼을 가지고 있는지 안 가지고 있는지에 따라 해당 칸의 steps를 따로 저장한다.

boj-17836 공주님을 구해라!

- 2개의 큐 q,q2를 사용하면 문제를 풀었다.
- 총 이동한 시간을 계속 세어 주어야하기 때문에 이런 방식을 차용(tuple 혹은 struct에 이동한 시간을 추가하여도 풀 수 있다)
- q에는 현재 steps시간만큼 이동한 칸들이 들어있고 각각에 대해서 q.front()하면서 참조하고 상하좌우 체크하면서 들어오는 새로운 칸은 다음 루프(steps+1)에서 확인하기 위해서 따로 임시로 q2에 넣어준다.
- 그리고 모든 q의 칸들(steps시간 만큼 이동한 칸)을 확인했으면 q2에 들어있는 칸(steps+1시간 만큼 이동한 칸)과 바꾸어주어 steps+1시간 만큼 이동한 칸에 대해서도 동일한 작업을 수행한다.

boj-17836 공주님을 구해라!

- https://github.com/alps-jbnu/ALPS_2020_Summer_Study/blob/master/KiwanKim/17836.cpp

boj-4991 로봇 청소기 (난이도 : 상)

https://www.acmicpc.net/problem/4991

boj-4991 로봇 청소기

- 저번주 달이 차오른다 가자! 문제와 거의 흡사한 문제
- 쓰레기에 번호를 부여하여 지금까지 치운 쓰레기를 비트마스크를 통해 기록한다.

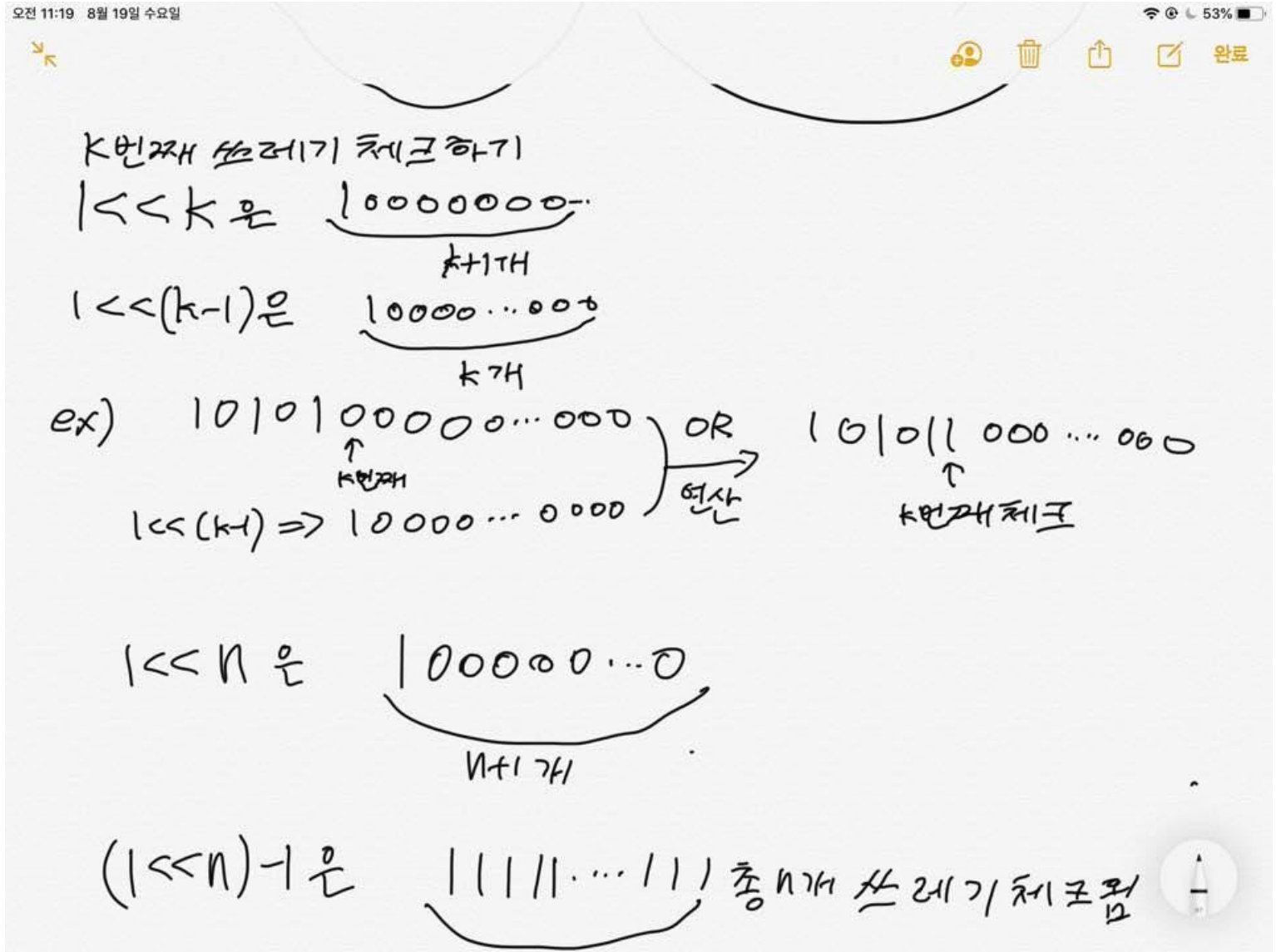
boj-4991 로봇 청소기

- 계속 이동한 칸수(steps)를 알고 있어야하기 때문에 q,q2 총 2개의 큐를 사용
- 각각의 문자들을 입력받으면 쓰레기를 만났을 때 비트마스크가 간단하도록 숫자로 치환하여 저장
- 쓰레기들은 1~N까지 각각의 번호를 갖고있다.
- struct status에는 x,y,trash(지금까지 치운 쓰레기 기록)이 포함
- cnt에는 해당 칸을 방문한 최소 steps를 저장해놓는다.
- cnt[x][y][지금까지 치운 쓰레기 기록(비트마스크)]

boj-4991 로봇 청소기(비트마스크)

- trash = trash | (1<<(room[tx][ty]-1));
- 해당 쓰레기를 만났을 때 번호에 해당하는 칸을 체크한다.
- Ex) 000100 -> 2번 쓰레기 만남 -> (000100) | (1<<(2-1)) -> 000110
- (이때 -1해주는 그림은 따로 첨부해놓음)
- trash==((1<<trash_cnt)-1)
- 모든 쓰레기를 치웠을 때 ret을 업데이트
- Ex) 111111 -> 총 6개의 쓰레기를 모두 치움 -> 111111==(1<<6 – 1)
- (이때 -1해주는 그림은 따로 첨부해놓음)

boj-4991 로봇 청소기(비트마스크)



boj-4991 로봇 청소기

- https://github.com/alps-jbnu/ALPS_2020_Summer_Study/blob/master/KiwanKim/4991.cpp

boj-1260 DFS와 BFS (난이도 : 하)

https://www.acmicpc.net/problem/1260

boj-1260 DFS와 BFS

- 시작 노드가 정해져 있음
- 그 노드와 연결된 모드 노드를 방문
- 같은 depth 안에서는 숫자 오름차순으로 방문해야 함
  - 미리 정렬을 해야 함

boj-1260 DFS와 BFS

- https://github.com/alps-jbnu/ALPS_2020_Summer_Study/blob/master/Jubin/week7/boj-1260-try-ac.cpp

boj-5014 스타트링크 (난이도 : 중)

https://www.acmicpc.net/problem/5014

boj-5014 스타트링크

- DFS로는 시간 초과 발생. BFS로 해결하기
- Pair의 first에는 현재 위치, second에는 엘리베이터 버튼 누른 횟수를 저장
- 이것을 queue에 넣음

boj-5014 스타트링크

- https://github.com/alps-jbnu/ALPS_2020_Summer_Study/blob/master/Jubin/week7/boj-5014.cpp

boj-2206 벽 부수고 이동하기 (난이도 : 상)

https://www.acmicpc.net/problem/2206

boj-2206 벽 부수고 이동하기

- 보통 미로찾기 문제는 visited를 2차원 배열만 써도 가능
- 그러나 아래의 두 가지 경우 때문에 3차원 배열을 사용
  - 벽을 부쉈는가
  - 벽을 안 부쉈는가
- 방문과 동시에 visited 처리를 잘 해야 함
- 벽을 만난 경우
  - 이미 벽을 1번 부순 이력이 있다면, 통과할 수 없음
  - 벽을 부순 이력이 없으면 벽을 부숴서 통과
- 땅을 만난 경우
  - 그대로 통과

boj-2206 벽 부수고 이동하기

- https://github.com/alps-jbnu/ALPS_2020_Summer_Study/blob/master/Jubin/week7/boj-2206.cpp

다음 8주차에는..

- Dynamic Programming을 배웁니다.