

[프로그래머스]크레인 인형뽑기 게임

▼ 상태	진행 중
📅 날짜	@2022년 6월 7일
≡ 공부유형	스터디
≡ 알고리즘	
▼ 사이트	프로그래머스
☑ 깃허브	<input type="checkbox"/>

코딩테스트 연습 - 크레인 인형뽑기 게임

게임개발자인 "조르디"는 크레인 인형뽑기 기계를 모바일 게임으로 만들려고 합니다. "조르디"는 게임의 재미를 높이기 위해 화면 구성과 규칙을 다음과 같이 게임 로직에 반영하려고 합니다. 게임 화면은 "1 x

🔗 <https://programmers.co.kr/learn/courses/30/lessons/64061>



개념

- 2차원 배열의 행 길이 구하기
 - `int N = board.length;`
- 자바 스택

```
import java.util.Stack;
Stack<Integer> stack = new Stack<>(); // int 형 스택 선언
stack.push(1);      // stack에 값 추가
stack.pop();        // stack 값 제거 ( 맨 위 )
stack.clear();      // stack 값 전체 제거
stack.peek();       // stack 맨 위 값 제거는 아니고 출력
stack.size();       // stack 크기 출력
stack.empty();      // stack 비어있으면 true 반환
stack.contains(1);  // stack에 1이 있는지 확인. 있으면 true
```

풀이

- // 5x5<= board <= 30x30
// 0<= 인형 종류 <=100
// 1<= moves 배열 크기 <=1000
- // 바구니 비어있지 않았으면 위를 탐색하고 넣기
// 맨위 a 빼서 같으면 a 빼기 + (answer 값 +2)
// 맨위 a 빼서 다르면 b 넣기
// 바구니 비어있으면 그냥 넣기

코드

```
import java.util.*;

public class 크레인인형뽑기게임 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int [][] b = {{0,0,0,0,0},{0,0,1,0,3},{0,2,5,0,1},{4,2,4,4,2},{3,5,1,3,1}};
        int [] m = {1,5,3,5,1,2,1,4};
        solution(b, m);
    }

    public static int solution(int[][] board, int[] moves) {
        int answer = 0;
        int N = board.length;
        Stack<Integer> basket = new Stack<>();

        // moves에서 하나씩 빼서 stack 에 넣기
        for (int j : moves) {
            j = j-1;
            for (int i = 0; i < N; i++) { // board의 세로방향으로
                if(board[i][j] !=0) { // 0이 아닌 인형이 있을 때까지 탐색
                    // 바구니에서 처리하기
                    if(!basket.empty()) { // 바구니가 비어있지 않다면
                        if(basket.peek() == board[i][j]) { // 맨 위의 것과 같으면
                            basket.pop(); // 바구니에서 빼버리기
                            answer += 2; // 없어진 인형을 2개 다 세줘야하므로 +2
                        }else { // 맨위의 것과 같지 않으면
                            basket.push(board[i][j]);
                        }
                    }else { // 바구니가 비어있으면
                        basket.push(board[i][j]); // 인형 넣기
                    }

                    board[i][j]=0; // 0으로 빈칸 만들기
                    break;
                }
            }
        }

        return answer;
    }
}
```

```
}  
}
```