리트코드 - 부분명령의 실행

https://leetcode.com/problems/execution-of-all-suffix-instructions-staying-in-a-grid/

정답코드

```
class Solution {
    public int[] executeInstructions(int n, int[] startPos, String s) {
        int[] answer = new int[s.length()];
        int[] dx = {-1, 0, 1, 0};
        int[] dy = {0, 1, 0, -1};
        int[] dir = {'U', 'R', 'D', 'L'};
        int nx = 0, ny = 0;
        for(int i = 0; i < s.length(); i++){
            int x = startPos[0];
            int y = startPos[1];
            int cnt = 0;
            for(int j = i; j < s.length(); j++){
                for(int k = 0; k < 4; k++){
                    if(s.charAt(j) == dir[k]){
                        nx = x + dx[k];
                        ny = y + dy[k];
                    }
                }
                if(nx < 0 || nx >=n || ny < 0 || ny >=n) break;
                cnt++;
                x = nx;
                y = ny;
            }
            answer[i]= cnt;
        }
        return answer;
    }
}
```

리트코드 - 최소 수도꼭지

https://leetcode.com/problems/minimum-number-of-taps-to-open-to-water-a-garden/

정답코드

```
class Solution {
    public int minTaps(int n, int[] ranges) {
        int answer = 0;
        int[][] line = new int[n + 1][2];
        for(int i = 0; i <= n; i++){
            line[i][0] = Math.max(0, i - ranges[i]);
            line[i][1] = Math.min(n, i + ranges[i]);
        }
        Arrays.sort(line, (a, b) -> a[0] - b[0]);
        int start = 0, end = 0, i = 0;
        while(i <= n){
            while(i <= n && line[i][0] <= start){
                end = Math.max(end, line[i][1]);
                i++;
            }
            answer++;
            if(start == end) return -1;
            if(end == n) return answer;
            start = end;
        }
        return answer;
    }
}
```

프로그래머스 Lv.1 - 달리기 경주
https://school.programmers.co.kr/learn/courses/30/lessons/178871

정답코드

```
import java.util.*;
class Solution {
    public String[] solution(String[] players, String[] callings) {
        String[] answer = new String[players.length];
        HashMap<String, Integer> sH = new HashMap<>();
        HashMap<Integer, String> nH = new HashMap<>();
        for(int i = 0; i < players.length; i++){
            sH.put(players[i], i);
            nH.put(i, players[i]);
        }
        for(String x : callings){
            int idx = sH.get(x);
            String pre = nH.get(idx-1);
            sH.put(x, idx-1);
            sH.put(pre, idx);
            nH.put(idx, pre);
            nH.put(idx-1, x);
        }
        for(int key : nH.keySet()){
            answer[key] = nH.get(key);
        }
        return answer;
    }
}
```

프로그래머스 Lv.1 - 숫자 짝궁
https://school.programmers.co.kr/learn/courses/30/lessons/131128

정답코드

```
import java.util.*;
class Solution {
    public String solution(String X, String Y) {
        StringBuilder answer = new StringBuilder();
        HashMap<Character, Integer> XH = new HashMap<>();
        HashMap<Character, Integer> YH = new HashMap<>();
        ArrayList<Character> al = new ArrayList<>();
        for(char x : X.toCharArray()){
            XH.put(x, XH.getOrDefault(x, 0) + 1);
        }
        for(char x : Y.toCharArray()){
            YH.put(x, YH.getOrDefault(x, 0) + 1);
        }
        for(char key : XH.keySet()){
            if(YH.containsKey(key)){
                int m = Math.min(XH.get(key), YH.get(key));
                for(int i = 0; i < m; i++) al.add(key);
            }
        }
        al.sort((a, b) -> b - a);
        if(al.size() == 0) return "-1";
        if(al.get(0) == '0') return "0";
        for(char x : al) answer.append(x);
        return answer.toString();
    }
}
```

프로그래머스 Lv.1 - 추억 점수
https://school.programmers.co.kr/learn/courses/30/lessons/176963

정답코드

```
import java.util.*;
class Solution {
    public int[] solution(String[] name, int[] yearning, String[][] photo) {
        int[] answer = new int[photo.length];
        HashMap<String, Integer> sH = new HashMap<>();
        for(int i = 0; i < name.length; i++){
            sH.put(name[i], yearning[i]);
        }
        for(int i = 0; i < photo.length; i++){
            int score = 0;
            for(String x : photo[i]){
                score += sH.getOrDefault(x, 0);
            }
            answer[i] = score;
        }
        return answer;
    }
}
```

프로그래머스 Lv.1 - 크레인 인형 뽑기
https://school.programmers.co.kr/learn/courses/30/lessons/64061

정답코드

```
import java.util.*;
class Solution {
    public int solution(int[][] board, int[] moves) {
        int answer = 0;
        Stack<Integer> stack = new Stack<>();
        for(int pos : moves){
            for(int i = 0; i < board.length; i++){
                if(board[i][pos-1] != 0){
                    int tmp = board[i][pos-1];
                    board[i][pos-1] = 0;
                    if(!stack.isEmpty() && stack.peek() == tmp){
                        stack.pop();
                        answer += 2;
                    }
                    else stack.push(tmp);
                    break;
                }
            }
        }
        return answer;
    }
}
```

프로그래머스 - 외톨이 알파벳
https://school.programmers.co.kr/learn/courses/15008/lessons/121683?language=java

정답코드

```
import java.util.*;
class Solution {
    public String solution(String input_string) {
        String answer = "";
        HashMap<Character, Integer> sH = new HashMap<>();
        sH.put(input_string.charAt(0), 1);
        for(int i = 1; i < input_string.length(); i++){
            if(input_string.charAt(i-1) != input_string.charAt(i)){
                sH.put(input_string.charAt(i), sH.getOrDefault(input_string.charAt(i), 0) + 1);
            }
        }
        StringBuilder sb = new StringBuilder();
        for(char key : sH.keySet()){
            if(sH.get(key) >= 2) sb.append(key);
        }
        answer = sb.toString();
        if(answer.equals("")) return "N";
        char[] tmp = answer.toCharArray();
        Arrays.sort(tmp);
        answer = String.valueOf(tmp);
        return answer;
    }
}
```