

1. Ants (BOJ 18295)

- 겹치지 않는 선에서 개미의 tag 값 할당하는 문제 (자연수)
- ex) [1 -1 0 3 10] 이라면 -1과 0 제외한 자연수들은 visit 배열에 true 저장
- visit 배열을 1부터 읽으면서 false인 최소값 출력했는데 30%에서 런타임 에러(!?)
MAX = 1000005
- 문제 이해를 제대로 못한느낌
(The smallest natural number that does not belong to the set)

```
int n, t, cnt;
set<int> a;
bool check[MAX];
int main() {
    cin >> n;
    for (int i = 0; i < n; i++) {
        cin >> t;
        if ((a.find(t) != a.end()) || t <= 0) {
            cnt++;
        }
        else {
            a.insert(t);
            check[t] = true;
        }
    }
    for (int i = 1; i <= n; i++) {
        if (!check[i]) {
            cout << i << '\n';
            break;
        }
    }
}
```

2. 탈옥 (BOJ 9376)

- <https://rebas.kr/770> 풀이법 보고 풀었습니다.
- <https://www.acmicpc.net/problem/9328> 와 같이 bfs 한번으로 3차원 visit 배열을 만들어 풀 수 있지 않을까 생각했지만 실패
- 위 블로그의 풀이법 말고 좀더 효율적인 풀이법이 있을지 궁금합니다.

- 까다로웠던 케이스

```
5 11
*#*****
*$*...*...*
*$*...*...*
*...*...*...*
*****.*
```

3. 배열의 분할(SWEA 9015)

- 단조증가와 단조감소 구분하는 영역을 구하는 문제

```
// n이 1일때 예외처리해야함.
if (n == 1) cout << '#' << j + 1 << ' ' << 1 << '\n';

for (int j = 0; j < n-1;) {
    // 시작점으로부터 단조증가하는 경우
    if (ary[j] <= ary[j + 1]) {
        // 증가하는 지점의 끝까지 인덱스 증가
        while (j < n && ary[j] <= ary[j + 1]) j++;
        j++;
        res++;
    }
    // 단조감소하는 경우라면
    else {
        // 감소하는 지점의 끝까지 인덱스 증가
        while (j < n && ary[j] >= ary[j + 1]) j++;
        j++;
        res++;
    }
}
cout << "#" << j + 1 << ' ' << res << '\n';
```

ex) 1 2 4 3 3 1의 경우 (j = 0부터 시작)
- j = 0 ~ 2까지 단조증가이므로 res++
하지만 j = 3일때는 ary[3] == ary[4]이므로
단조감소인지 단조증가인지 판단할 수가
없어 막힌 문제입니다..

- n이 10만이면 nlogn까지 풀이가 가능한데 dp라고 가정해도 점화식을 세우기 상당히 까다롭습니다.