



6주차 삼성A형 기출

삼성 A형?



실제 삼성 코딩테스트 수준의 문제 2개



1sol : A 취득 (서류 우대)



2sol : A+ 취득 (서류, 코테 우대)



브루트포스, 완전탐색, 백트래킹 등의 구현 위주



문제를 절대 대충 읽지 말고 정독!!!

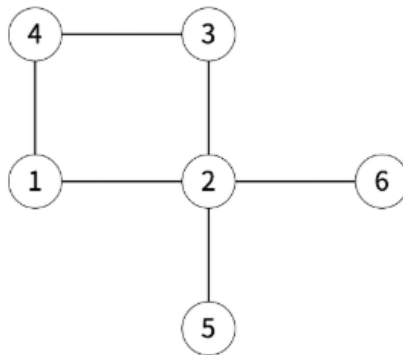
17471_게리맨더링

문제

백준시의 시장 최백준은 지난 몇 년간 게리맨더링을 통해서 자신의 당에게 유리하게 선거구를 획정했다. 견제할 권력이 없어진 최백준은 권력을 매우 부당하게 행사했고, 심지어는 시의 이름도 백준시로 변경했다. 이번 선거에서는 최대한 공정하게 선거구를 획정하려고 한다.

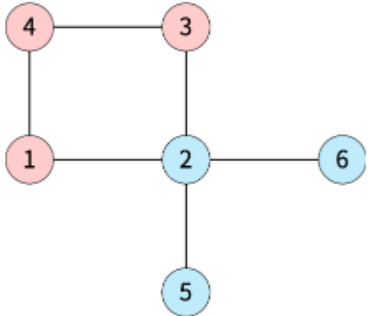
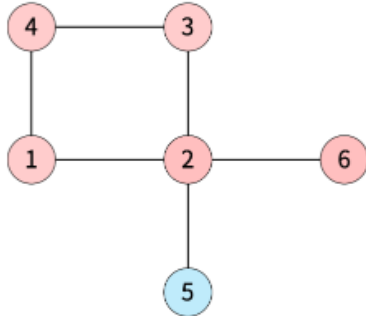
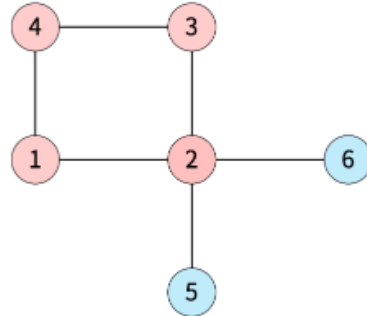
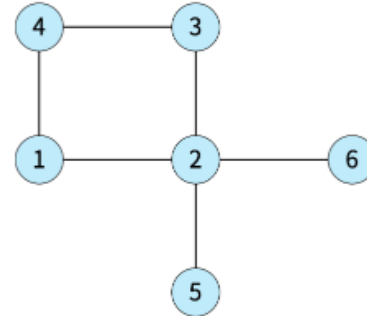
백준시는 N 개의 구역으로 나누어져 있고, 구역은 1번부터 N 번까지 번호가 매겨져 있다. 구역을 두 개의 선거구로 나눠야 하고, 각 구역은 두 선거구 중 하나에 포함되어야 한다. 선거구는 구역을 적어도 하나 포함해야 하고, 한 선거구에 포함되어 있는 구역은 모두 연결되어 있어야 한다. 구역 A에서 인접한 구역을 통해서 구역 B로 갈 수 있을 때, 두 구역은 연결되어 있다고 한다. 중간에 통하는 인접한 구역은 0개 이상이어야 하고, 모두 같은 선거구에 포함된 구역이어야 한다.

아래 그림은 6개의 구역이 있는 것이고, 인접한 구역은 선으로 연결되어 있다.



아래는 백준시를 두 선거구로 나눈 4가지 방법이며, 가능한 방법과 불가능한 방법에 대한 예시이다.

아래는 백준시를 두 선거구로 나눈 4가지 방법이며, 가능한 방법과 불가능한 방법에 대한 예시이다.

			
<p>가능한 방법</p> <p>[1, 3, 4]와 [2, 5, 6]으로 나누어져 있다.</p>	<p>가능한 방법</p> <p>[1, 2, 3, 4, 6]과 [5]로 나누어져 있다.</p>	<p>불가능한 방법</p> <p>[1, 2, 3, 4]와 [5, 6]으로 나누어져 있는데, 5와 6이 연결되어 있지 않다.</p>	<p>불가능한 방법</p> <p>각 선거구는 적어도 하나의 구역을 포함해야 한다.</p>

공평하게 선거구를 나누기 위해 두 선거구에 포함된 인구의 차이를 최소화하려고 한다. 백준시의 정보가 주어졌을 때, 인구 차이의 최솟값을 구해보자.

입력

첫째 줄에 구역의 개수 N 이 주어진다. 둘째 줄에 구역의 인구가 1번 구역부터 N 번 구역까지 순서대로 주어진다. 인구는 공백으로 구분되어져 있다.

셋째 줄부터 N 개의 줄에 각 구역과 인접한 구역의 정보가 주어진다. 각 정보의 첫 번째 정수는 그 구역과 인접한 구역의 수이고, 이후 인접한 구역의 번호가 주어진다. 모든 값은 정수로 구분되어져 있다.

구역 A가 구역 B와 인접하면 구역 B도 구역 A와 인접하다. 인접한 구역이 없을 수도 있다.

출력

첫째 줄에 백준시를 두 선거구로 나누었을 때, 두 선거구의 인구 차이의 최솟값을 출력한다. 두 선거구로 나눌 수 없는 경우에는 -1을 출력한다.

예제 입력 1 복사

```
6
5 2 3 4 1 2
2 2 4
4 1 3 6 5
2 4 2
2 1 3
1 2
1 2
```

선거구를 [1, 4], [2, 3, 5, 6]으로 나누면 각 선거구의 인구는 9, 8이 된다. 인구 차이는 1이고, 이 값보다 더 작은 값으로 선거구를 나눌 수는 없다.

예제 입력 2 복사

```
6
1 1 1 1 1 1
2 2 4
4 1 3 6 5
2 4 2
2 1 3
1 2
1 2
```

선거구를 [1, 3, 4], [2, 5, 6]으로 나누면 인구 차이가 0이다.

예제 출력 1 복사

1

예제 출력 2 복사

0

17471_게리맨더링

노트

게리맨더링은 특정 후보자나 정당에 유리하게 선거구를 획정하는 것을 의미한다.

출처

- 문제를 만든 사람: baekjoon

알고리즘 분류

- 그래프 이론
- 그래프 탐색
- 브루트포스 알고리즘

조합에서의 next_permutation 활용

```
np.clear();  
np.resize(n);  
for (int j = 0; j < i; j++) {  
    np[n - j - 1] = 1;  
}
```

0과 1로 np 벡터를 채움

$v = \{ 3, 5, 2, 7, 9, 10, 8 \}$

$np = \{ 0, 0, 0, 0, 0, 1, 1 \}$



현재 1, 2, 3, 4, 5번과 6, 7번으로 선거구를 나눈 상태
next_permutation을 반복하여 완전탐색! (브루트포스)

```
do {  
    // 매번 조건에 만족하는지 검토 -> 정답 update  
} while (next_permutation(np.begin(), np.end()));
```

17070_파이프 옮기기1

- DP라는 것을 알아내는게 중요
- 가로, 세로, 대각 세가지 경우에 따른 점화식 설계

5 17070번 제출 맞은 사람 숏코딩 재채점 채점 현황 내 제출 난이도 기여 강의▼ 질문 검색

파이프 옮기기 1 성공 분류

5 Gold V

다이나믹 프로그래밍

난이도 제공: solved.ac — 난이도 투표하러 가기

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초 (추가 시간 없음)	512 MB	11190	5493	3655	49.405%

17070_파이프 옮기기1(가로 파이프)

```
if (k == 0) { // 가로
    if (j + 1 > n) // 범위 체크
        continue;
    if (arr[i][j + 1] != 1) { // 벽이 아닐경우 진행 가능
        dp[i][j + 1][0] += dp[i][j][0]; // 가로
        if (arr[i + 1][j] != 1 && arr[i + 1][j + 1] != 1) // 대각
            dp[i + 1][j + 1][1] += dp[i][j][0];
    }
}
```

17070_파이프 옮기기1(세로 파이프)

```
else if (k == 2) { // 세로
    if (i + 1 > n) // 범위 체크
        continue;
    if (arr[i + 1][j] != 1) { // 벽이 아닐경우 진행 가능
        dp[i + 1][j][2] += dp[i][j][2]; // 세로
        if (arr[i][j + 1] != 1 && arr[i + 1][j + 1] != 1) // 대각
            dp[i + 1][j + 1][1] += dp[i][j][2];
    }
}
```

17070_파이프 옮기기1(대각 파이프)

```
else if (k == 1) { // 대각 : 가로, 세로, 대각 모두 설치 가능
    bool flag1 = false, flag2 = false;
    if (j + 1 <= n && arr[i][j + 1] != 1) { // 가로
        dp[i][j + 1][0] += dp[i][j][1];
        flag1 = true;
    }
    if (i + 1 <= n && arr[i + 1][j] != 1) { // 세로
        dp[i + 1][j][2] += dp[i][j][1];
        flag2 = true;
    }

    // 가로와 세로 모두 가능하며 대각에도 파이프 설치가 가능할 경우
    if (flag1 && flag2 && arr[i + 1][j + 1] != 1)
        dp[i + 1][j + 1][1] += dp[i][j][1];
}
```

17136_색종이 붙이기

2 17136번

제출

맞은 사람

숏코딩

재채점

채점 현황

내 제출

난이도 기여

강의

질문 검색

색종이 붙이기

성공

분류

2 Gold II

백트래킹 브루트포스 알고리즘

난이도 제공: solved.ac — 난이도 투표하러 가기

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	512 MB	13073	4807	2384	32.255%

17136_색종이 붙이기

- 전체적인 논리 설계 후 코드 작성 시작
- 코드가 복잡해질 땐 함수를 사용하여 분할해라!
 - * 디버깅이 편해지고 가독성이 높아짐

17136_색종이 붙이기(main)

```
66 int main() {
67     cin.tie(NULL);
68     ios::sync_with_stdio(false);
69
70     for (int i = 1; i <= 10; i++) {
71         for (int j = 1; j <= 10; j++) {
72             cin >> arr[i][j];
73         }
74     }
75
76     go(1, 0);
77     if (ans == 1e9)
78         cout << -1;
79     else
80         cout << ans;
81
82
83     return 0;
84 }
```

17136_색종이 붙이기(핵심 재귀함수)

```
39 void go(int idx, int cnt) {
40     if (idx == 101) { // 모든 배열을 돌고난 후 정답 갱신
41         ans = min(ans, cnt);
42         return;
43     }
44     // idx는 y좌표와 x좌표 두 정보를 모두 담고있음
45     int cury = (idx - 1) / 10 + 1;
46     int curx = ((idx - 1) % 10) + 1;
47
48     // 색종이 붙일필요 없으면 그대로 다음 좌표 진행
49     if (arr[cury][curx] <= 0) {
50         go(idx + 1, cnt);
51     }
52
53     else {
54         for (int i = 5; i >= 1; i--) { // i는 색종이 한 변의 길이
55             if (paper_cnt[i] == 0) // 색종이가 남아있니?
56                 continue;
57             if (isPos(cury, curx, i)) { // 해당 좌표에 색종이 붙이기 가능?
58                 func(cury, curx, i, -1); // 색종이를 붙임
59                 go(idx + 1, cnt + 1);
60                 func(cury, curx, i, 1); // 색종이를 땀
61             }
62         }
63     }
64 }
```

17136_색종이 붙이기(보조 함수)

```
16 bool isPos(int y, int x, int l) {  
17     // y, x를 왼쪽 상단으로 했을때 변이 1인 색종이를 붙일수 있니?  
18     // 있으면 true 없으면 false를 리턴  
19     for (int i = y; i < y + l; i++) {  
20         for (int j = x; j < x + l; j++) {  
21             if (i > 10 || j > 10)  
22                 return false;  
23             if (arr[i][j] <= 0)  
24                 return false;  
25         }  
26     }  
27     return true;  
28 }  
29  
30 void func(int y, int x, int l, int what) {  
31     paper_cnt[l] += what; // 색종이의 개수 업데이트(감소 or 증가)  
32     for (int i = y; i < y + l; i++) {  
33         for (int j = x; j < x + l; j++) {  
34             arr[i][j] += what;  
35         }  
36     }  
37 }
```


삼성 코딩테스트 대비!

BAEKJOON
ONLINE JUDGE

k95yong | 0 코인 | 설정 | 로그아웃 | 27 | ★

문제 ▾

문제집

대회

채점 현황

랭킹

게시판

그룹

블로그

강의



Microsoft 365



Microsoft 365 Business Basic

250명까지 가능한 화상 회의, 맞춤형 도메인을 통한 이메일 호스팅이 비즈니스를 더욱 빠르고 민첩하게

지금 구입하기 →

인기

code.plus

시리즈

공개

비공개

북마크

좋아요

신고

내 문제집

만들기

번호	만든 사람	제목	진행도
1152	baekjoon	삼성 SW 역량 테스트 기출 문제	28 / 37
2771	baekjoon	삼성 A형 기출 문제	8 / 9