문제 A3: [SOL] 사과나무

실행시간 제한: 1 Sec 메모리사용 제한: 128 MB
제출: 0 통과: NAN%
[제출]

문제 설명

최소 5에서 10 인 사각형 모양의 map에서 1은 사과나무 2는 화재발생을 의미한다.
화재는 상하좌우 번지며, 화재 지점의 개수는 제한이 없다. 나무를 제거하면 자른 나무는 구할 수 없지만 더 이상 화재가 번지는 것을 막는다.
아래 예시에서 위에 있는 나무와 아래 있는 나무를 제거하면 5개의 나무가 남는다.
그림에서 빨강색으로 표시된 것이 화재 지점이며, 주황색이 제거된 사과 나무, 초록색이 남은 사과나무를 의미한다.

0	1	1	1	1	0
0	0	2	0	1	0
0	0	1	0	0	0
0	0	1	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

나무는 제거 안 할 수도 1개만 할 수도 있으며, 2개까지 제거 할 수 있다.

```
#include <stdio.h>

#define MAX (10+2)
typedef struct _ind
{
	int r;
	int c;
}IND;
int N;
int map[MAX][MAX];
int end;
IND Que[MAX*MAX];
int swp;
int dr[] = { 0, -1, 0, 1 };
int dc[] = { 1, 0, -1, 0 };
int min;
int count;
void input(void)
{
	register int i, j;
	int tmp;
	scanf("%d", &N);
	for (i = 0; i < N; i++)
	{
		for (j = 0; j < N; j++)
		{
			scanf("%d", &tmp);
			if (tmp == 2)
				Que[swp++] = (IND){ i, j };
			else if (tmp == 1)
				count++;
			map[i][j] = tmp;
		}
	}
	end = N*N;
}

int BFS(int add)
{
	int visit[MAX][MAX] = { 0 };
	int wp, rp;
	int cr, cc;
	register int i, nr, nc;
	IND out;
	wp = swp;
	rp = 0;
	while (wp > rp)
	{
		out = Que[rp++];
		cr = out.r;
		cc = out.c;
		visit[cr][cc] = 1;
		for (i = 0; i < 4; i++)
		{
			nr = cr + dr[i];
			nc = cc + dc[i];
			if (nr < 0 || nc <0 || nr >= N || nc >= N)	continue;
			if (map[nr][nc] == 1 && !visit[nr][nc])
			{
				visit[nr][nc] = 1;
				Que[wp++] = (IND){ nr, nc };
				add++;
				if (add >= min) return min;
			}
		}
	}
	return add;
}
void DFS(int depth,int now,int en)
{
	int i, j;
	if (now == en)
	{
		min = BFS(en);
		return;
	}
	if (depth >= end)	return;
	i = depth / N;
	j = depth % N;
	if (map[i][j] == 1)
	{
		map[i][j] = 0;
		DFS(depth + 1, now+1,en);
		map[i][j] = 1;
	}
	DFS(depth + 1, now,en);
}
int main(void)
```

```
{
	input();

	min = 0x7fff0000;
	if (!swp)
		printf("%d", count);
	else
	{
		for (int i = 0; i < 3; i++)
		{
			if (i > count)  break;
			DFS(0, 0, i);
		}
		printf("%d", count - min);
	}
	return 0;
}
```

입력 설명

첫 줄에는 map의 크기 N이 입력된다. (5<= N <=10)
두번 째 줄부터 N줄에 걸쳐 사과나무와 화재 지점에 대한 정보가 입력된다.
사과나무는 1, 화재 지점은 2를 의미하며 화재 지점의 개수는 제한이 없다.

출력 설명

화재로 소실 되지 않고 남을수 있는 사과나무의 최대 수를 출력한다.

입력 예시

```
6
0 1 1 1 1 0
0 0 2 0 1 0
0 0 1 0 0 0
0 0 1 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
```

출력 예시

```
5
```

[제출]