

1. 문제

정부에서 수출산업공단 계획을 발표하였다. 이 때 모든 부지에 건물을 짓는 것이 아니라 자연환경을 보호하면서 공단을 조성할 것이라고 한다.

이를 위해 정부에서는 몇 가지 원칙을 가지고 수출산업공단 계획을 추진할 계획이다.

- 부지의 외곽지역의 구획은 모든 자연환경을 위해 공원으로 조성한다.
 - 구획 하나당 한 개의 건물만 세울 수 있다.
 - 건물의 층수는 상하좌우에 공원이 인접해 있을 경우 공원의 자연 보호를 위해 높이를 1층으로 제한한다.
 - 건물이 상하좌우 어느 곳에서도 인접하지 않을 경우(대각선은 제외) 건물의 층수는 조건에 따라 높게 세울 수 있다. 건물의 높이는 자신(해당 부지)을 포함하여 상하좌우에 있는 건물의 개수로 결정한다.

이 때 수출산업공단을 세울 부지의 크기는 가로, 세로의 길이가 L(L은 자연수)인 정사각형 형태이고, 가로, 세로의 길이가 1인 구획으로 부지를 나눈다는 가정하에 공단에 입주하는 각 건물들의 층 수를 구하라

예를 들어 아래 그림은 L이 9인 부지에 구획을 나눈 것으로 외곽지역은 공원(P)이 구성되어 있고, 내부에는 공원과 건물이 같이 구성되어 있다.

P	P	P	P	P	P	P	P	P
P	P	P	B	P	P	P	P	P
P	P	P	B	B	B	B	P	P
P	B	B	B	B	B	B	P	P
P	P	P	B	B	B	B	B	P
P	P	P	B	P	B	B	B	P
P	B	P	P	P	B	P	P	P
P	B	B	P	B	P	B	P	P
P	P	P	P	P	P	P	P	P

[그림]

수출 산업 공단의 지도

지도에서 나타난 것처럼 공원 구획은 P로 표시하고, 건물이 세워질 구획은 B로 표시하였다. 서울시 발표 내용을 보면 동그라미로 표시가 된 구획은 왼쪽에 공원(P)이 인접해 있기 때문에 건물의 높이가 1이 된다 사각형으로 표시된 구획의 건물은 왼쪽(3개), 오른쪽(1개), 위쪽(2개), 아래쪽(2개), 자기 자신(1개)을 포함하여 총 9개가 되어 9층의 건물을 세울 수 있다.

(1) 문제 제시

수출 산업공단 계획에 맞게 건물이 세워질 때 모든 건물의 높이(층수)의 합을 계산하는 프로그램을 작성하라. 또한 입력된 높이, 즉 특정 높이 이상의 건물 개수를 구하라.

(2) 제약 사항

부지의 크기 n 은 $3 < n < 15$ 인 자연수이다. 층수가 h 이상인 건물을 구하기 위해 3보다 큰 정수를 입력한다($3 < h < 15$)

2. 입력 및 출력 형태

(1) 입력

첫 째 줄은 테스트케이스 N 을 입력한다($0 < N < 4$).

둘 째 줄은 부지의 크기(n)와 건물 높이(h)를 입력한다.

셋 째 줄 이후는 부지의 구획 정보를 건물은 B, 공원은 P로 표시된 2차원 배열로 입력한다. 그 다음은 테스트 케이스만큼 반복한다.

(2) 입력 예

3

9 5

PPPPPPPPP

PPPBPPPPP

PPPB BBBPP

PBBBBBBPP

PPPB BBBPP

PPPB PBBBBP

PBPPPBPBP

PBBPBPBPBP

PPPPPPPPP

9 10

PPPPPPPPP

PPBBPPPPP

PPPB BBBPP

PBBBBBBPP

PPPB BBBPP

PPPB PBBBBP

PBPPBBBPBP

PBBPBPBPBP

PPPPPPPPP

8 4

PPPPPPPPP

PPPB BPBP

PB BBBBP

PBBBPBBP

PB PBBBBP

PB PBPBBP

PBBPBPBP

PPPPPPPPP

(3) 출력

#x 형태로 시작하고 공백을 두고 그 다음에 결과를 출력한다.(x는 테스트케이스 번호).

전체 건물의 높이의 합을 출력하고 입력된 높이와 같거나 큰 층수를 가진 건물의 수를 출력한다.

(4) 출력 예

output.txt
#1 68 5
#2 80 3
#3 58 3

=====

문제

9x9 형태의 스도쿠는

3x3 형태의 그룹 9개로 구성되어 다음과 같은 규칙을 가지고 있다.

각 그룹은 정사각형 3x3의 형태를 가지며, 숫자 1~9까지 하나씩만 들어가야 한다.

가로줄에 숫자가 1~9까지 하나씩만 들어가야 한다.

세로줄에 숫자가 1~9까지 하나씩만 들어가야 한다.

정호는 휴식시간에 9x9의 스도쿠를 열심히 풀었지만 수현이가 스도쿠에 오류가 있다고 이야기했다. 정호는 자신이 푼 스도쿠가 어디에서 틀렸는지 위치를 찾고, 옳은 값을 찾고 싶어 한다.

(1) 문제 1

스도쿠를 잘 풀었는지 확인하고 틀린 속이 있으면 그 위치를 찾아 옳은 값을 출력하는 프로그램을 작성하라

4	5	2	6	1	8	3	7	9
3	7	1	4	2	9	8	6	5
9	8	1	5	7	3	2	4	1
7	3	4	1	6	2	5	6	8
6	2	9	8	4	5	7	1	3
8	1	5	3	9	8	6	2	4
2	9	3	7	5	4	1	8	6
1	4	8	4	3	6	9	5	7
5	6	7	9	8	1	4	3	2

- (2) 제약사항
- 틀린 값은 5개 이내로 한다.
- 틀린 값이 같은 행이나 열 또는 그룹에 배치하지 않는다.

2. 입력 및 출력 형식

- (1) 입력
- 9x9 크기의 스도쿠 행렬을 입력한다.
- (2) 입력 예

input.txt								
			4	5	2	6	1	8
			3	7	1	4	2	9
			9	8	1	5	7	3
			7	3	4	1	6	2
			6	2	9	8	4	5
			8	1	5	3	9	8
			2	9	3	7	5	4
			1	4	8	4	3	6
			5	6	7	9	8	1

- (3) 출력
- 행이나 열을 검사하여 중복된 값을 찾고, 중복된 값 중에 잘못된 값의 위치를 출력한다.
- #x 형태로 시작하고 공백을 두고 그 다음에 위치와 옳은 값을 출력한다(첨자는 1부터 시작한다.)
- (4) 출력 예

output.txt								
#1	3	3	6					
#2	4	8	9					
#3	6	6	7					
#4	8	4	2					

=====

문제

$N \times N$ 크기의 공간에 물고기 M 마리와 아기 상어 1마리가 있다. 공간은 1×1 크기의 정사각형 칸으로 나누어져 있다. 한 칸에는 물고기가 최대 1마리 존재한다.

아기 상어와 물고기는 모두 크기를 가지고 있고, 이 크기는 자연수이다. 가장 처음에 아기 상어의 크기는 2이고, 아기 상어는 1초에 상하좌우로 인접한 한 칸씩 이동한다.

아기 상어는 자신의 크기보다 큰 물고기가 있는 칸은 지나갈 수 없고, 나머지 칸은 모두 지나갈 수 있다. 아기 상어는 자신의 크기보다 작은 물고기만 먹을 수 있다. 따라서, 크기가 같은 물고기는 먹을 수 없지만, 그 물고기가 있는 칸은 지나갈 수 있다.

아기 상어가 어디로 이동할지 결정하는 방법은 아래와 같다.

- 더 이상 먹을 수 있는 물고기가 공간에 없다면 아기 상어는 엄마 상어에게 도움을 요청한다.
- 먹을 수 있는 물고기가 1마리라면, 그 물고기를 먹으러 간다.
- 먹을 수 있는 물고기가 1마리보다 많다면, 거리가 가장 가까운 물고기를 먹으러 간다.
 - 거리는 아기 상어가 있는 칸에서 물고기가 있는 칸으로 이동할 때, 지나야하는 칸의 개수의 최솟값이다.
 - 거리가 가까운 물고기가 많다면, 가장 위에 있는 물고기, 그러한 물고기가 여러마리라면, 가장 왼쪽에 있는 물고기를 먹는다.

아기 상어의 이동은 1초 걸리고, 물고기를 먹는데 걸리는 시간은 없다고 가정한다. 즉, 아기 상어가 먹을 수 있는 물고기가 있는 칸으로 이동했다면, 이동과 동시에 물고기를 먹는다. 물고기를 먹으면, 그 칸은 빈 칸이 된다.

아기 상어는 자신의 크기와 같은 수의 물고기를 먹을 때 마다 크기가 1 증가한다. 예를 들어, 크기가 2인 아기 상어는 물고기를 2마리 먹으면 크기가 3이 된다.

공간의 상태가 주어졌을 때, 아기 상어가 몇 초 동안 엄마 상어에게 도움을 요청하지 않고 물고기를 잡아먹을 수 있는지 구하는 프로그램을 작성하시오.

입력

첫째 줄에 공간의 크기 $N(2 \leq N \leq 20)$ 이 주어진다.

둘째 줄부터 N 개의 줄에 공간의 상태가 주어진다. 공간의 상태는 0, 1, 2, 3, 4, 5, 6, 9로 이루어져 있고, 아래와 같은 의미를 가진다.

- 0: 빈 칸
- 1, 2, 3, 4, 5, 6: 칸에 있는 물고기의 크기
- 9: 아기 상어의 위치

아기 상어는 공간에 한 마리 있다.

출력

첫째 줄에 아기 상어가 엄마 상어에게 도움을 요청하지 않고 물고기를 잡아먹을 수 있는 시간을 출력한다.

예제 입력 1	예제 입력 2	예제 입력 3
3	3	4
0 0 0	0 0 1	4 3 2 1
0 0 0	0 0 0	0 0 0 0
0 9 0	0 9 0	0 0 9 0
예제 출력 1	예제 출력 2	예제 출력 3
0	3	14

예제 입력 4	예제 입력 5	예제 입력 6
6	6	6
5 4 3 2 3 4	6 0 6 0 6 1	1 1 1 1 1 1
4 3 2 3 4 5	0 0 0 0 0 2	2 2 6 2 2 3
3 2 9 5 6 6	2 3 4 5 6 6	2 2 5 2 2 3
2 1 2 3 4 5	0 0 0 0 0 2	2 2 2 4 6 3
3 2 1 6 5 4	0 2 0 0 0 0	0 0 0 0 0 6
6 6 6 6 6 6	3 9 3 0 0 1	0 0 0 0 0 9
예제 출력 4	예제 출력 5	예제 출력 6
60	48	39

=====

문제

길이가 N인 수식이 있다. 수식은 0보다 크거나 같고, 9보다 작거나 같은 정수와 연산자(+, -, ×)로 이루어져 있다. 연산자 우선순위는 모두 동일하기 때문에, 수식을 계산할 때는 왼쪽에서부터 순서대로 계산해야 한다. 예를 들어, $3+8\times7-9\times2$ 의 결과는 136이다.

수식에 괄호를 추가하면, 괄호 안에 들어있는 식은 먼저 계산해야 한다. 단, 괄호 안에는 연산자가 하나만 들어 있어야 한다. 예를 들어, $3+8\times7-9\times2$ 에 괄호를 $3+(8\times7)-(9\times2)$ 와 같이 추가했으면, 식의 결과는 41이 된다. 하지만, 중첩된 괄호는 사용할 수 없다. 즉, $3+((8\times7)-9)\times2$, $3+((8\times7)-(9\times2))$ 은 모두 괄호 안에 괄호가 있기 때문에, 올바른 식이 아니다.

수식이 주어졌을 때, 괄호를 적절히 추가해 만들 수 있는 식의 결과의 최댓값을 구하는 프로그램을 작성하시오. 추가하는 괄호 개수의 제한은 없으며, 추가하지 않아도 된다.

입력

첫째 줄에 수식의 길이 N($1 \leq N \leq 19$)가 주어진다. 둘째 줄에는 수식이 주어진다. 수식에 포함된 정수는 모두 0보다 크거나 같고, 9보다 작거나 같다. 문자열은 정수로 시작하고, 연산자와 정수가 번갈아가면서 나온다. 연산자는 +, -, * 중 하나이다. 여기서 *는 곱하기 연산을 나타내는 × 연산이다. 항상 올바른 수식만 주어지기 때문에, N은 홀수이다.

출력

첫째 줄에 괄호를 적절히 추가해서 얻을 수 있는 결과의 최댓값을 출력한다. 정답은 231보다 작고, -231보다 크다.

예제입력1	예제출력1	예제입력1	예제출력2
9 $3+8\times7-9\times2$	136	5 $8\times3+5$	64
예제입력3	예제출력3	예제입력4	예제출력4
7 $8\times3+5+2$	66	19 $1\times2+3\times4\times5-6\times7\times8\times9\times0$	0
예제입력5	예제출력5	예제입력6	예제출력6
19 $1\times2+3\times4\times5-6\times7\times8\times9\times9$	426384	1-9-1-9-1-9-1-9-1-9	24

=====

문제

승택이는 강을 건너려 한다.

승택이는 수영을 못하기 때문에, 강에 놓인 징검다리를 밟고 건너갈 것이다.

승택이는 수영은 못하지만 제자리뛰기는 정말 잘한다. 원하는 어느 곳으로든지 점프해서 바로 갈 수가 있다.

승택이는 이제 강의 한쪽 변 앞에 서 있다.

강엔 1번부터 시작해 2번, 3번, ..., N번 징검다리가 차례대로 놓여 있다.

강의 폭이 넓은 탓에 징검다리의 수는 엄청나게 많다.

이 징검다리를 모두 밟고 싶지는 않았던 승택이는 제자리뛰기 실력을 발휘해

적절한 개수의 징검다리만을 밟고 가기로 했다.

물론 강 건너편으로 바로 점프하는 것도 가능하지만, 더 재미있게 강을 건너기 위해 승택이는 다음과 같은 규칙을 정했다.

1. 첫 징검다리는 점프해서 아무 것이나 밟을 수 있다. 이 점프가 첫 점프이다.
2. 두 번째 점프부터는 이전에 점프한 거리보다 1 이상 더 긴 거리를 뛰어야만 한다.
3. N번 징검다리는 반드시 밟아야 한다.
4. N번 징검다리를 밟은 후 강 건너로 이동할 땐 점프를 하지 않으므로 위의 규칙이 적용되지 않는다.

승택이가 위의 규칙을 지키며 강을 건널 때, 밟을 수 있는 징검다리의 최대 수는 몇 개일까?

입력

첫째 줄에 테스트 케이스의 수 T가 주어진다.

각 테스트 케이스는 정수 한 개로 이루어져 있으며,

징검다리의 총 수 N을 의미한다. ($1 \leq N \leq 10^{16}$)

C, C++의 경우, N이 int 범위를 넘어갈 수 있음에 주의해야 한다.

자료형 long long을 사용하고, 입력과 출력을 %lld로 수행하면 된다.

출력

각 테스트 케이스마다 한 줄에 승택이가 밟을 수 있는 최대 징검다리 수를 출력한다.

예제 입력 1

4
1
2
100
1000000

예제 출력 1

1
1
13
1413

=====

1. 문제

시가전에서 적의 탱크와 대치하고 있는 아군의 탱크가 있다. 아군의 탱크가 포탄을 쏘기 위해 정지되어 있는 상태이다. 현재 아군의 탱크는 가로 나 세로에 있는 타겟만을 공격할 수 있다. 사방으로 벽이 있는 경우 벽 너머에 있는 타겟은 보이지 않기 때문에 공격할 수 없다. 마찬가지로 타겟 뒤에 있는 타겟은 앞에 타겟으로 인해 보이지 않기 때문에 공격할 수 없고 아군의 탱크도 공격하지 않는다.
가로 7칸 세로 7칸으로 이루어진 시가전 상황에서 T는 아군 탱크, E는 적군의 탱크(타겟), 건물은 B로 표시하되 가시성을 위해 색칠이 되어 있다.

	0	1	2	3	4	5	6
0	E		T		E	E	
1							
2	E		E			T	E
3				T			E
4	E	T	E				
5		E		E			T
6							E

아군의 탱크가 5대로 (0, 2), (2, 5), (3, 3), (4, 2), (5, 6)에 위치하고 있다. 첫 번째 있는 아군 탱크 (0, 2)는 적군 탱크 (0, 0), (0, 4)를 공격할 수 있고, 적군 탱크 (2, 2)는 건물 때문에 볼 수 없어 공격하지 않는다. 적군 탱크(0, 5)는 적군 탱크 (0, 4) 때문이다. 아군 탱크가 공격한 것은 다른 아군 탱크가 다시 공격하지 않는다.

(1) 문제 제시

상부에서 아군의 탱크가 공격하여 폭파시킨 적의 탱크의 수를 알아온다고 할 때 이를 구하기 위한 프로그램을 작성하라.

(2) 제약사항

시가전을 위한 지도를 2차원 배열로 입력한다. 지도의 크기는 10 X 10이하로 한다.

2. 입력 및 출력 형태

(1) 입력

첫째 줄은 테스트케이스 N을 입력한다. (1 < N ≤ 3).

둘 째 줄은 지도의 크기 n을 입력한다. 그 다음 줄은 2차원 배열을 입력하되 아군 탱크는 T로, 적군 탱크는 E로, 건물은 B로 표시하고, 비어 있는 공간은 0으로 표시한다.

테스트케이스만큼 지도의 크기와 2차원 배열을 추가로 입력한다.

(2) 입력 예

input.txt
2
7
E 0 T 0 E E 0
0 B B 0 B 0 0
E B E 0 T E
B B 0 T B 0 E
E T E 0 B 0 0
0 E 0 E B 0 T
0 0 B B B B E
5
E 0 T 0 E
0 B B 0 B
E B E 0 B
B B 0 T B
E T E 0 B

(3) 출력

#x형태로 시작하고 공백을 두고 그 다음에 군이 격추한 적의 탱크 수를 출력한다(x는 테스트케이스 번호)

(4) 출력 예

output.txt
#1 10
#2 4

=====

1. 문제

한 줄에 빨강, 파랑, 초록 색상의 공들이 섞여 있다. 각 공의 색깔은 순서대로 R, B, G로 표현한다. 여러분은 한 턴에 제일 앞의 공 혹은 제일 뒤의 공을 제거할 수 있다면 한 색깔의 공만 남기기 위해서는 최소 몇 번의 턴이 필요한지 구하라.

(1) 문제 제시

제일 앞의 공 혹은 제일 뒤의 공을 제거할 수 있다면 한 색깔의 공만 남기기 위해서는 최소 몇 번의 턴이 필요하지 구하라.

(2) 제약사항

한 줄에 있는 공의 수는 1이상 10,000이하이다.

2. 입력 및 출력 형태

(1) 입력

첫 째 줄에는 테스트 케이스 T를 입력한다.($1 \leq n \leq 10,000$)

둘 째 줄부터 하나의 테스트 케이스에 대한 문자열(S)을 입력한다. 입력된 문자열은 문자 R, G, B로 이루어지고, 그 길이는 1 이상 10,000이하이다.

(2) 입력 예

input.txt
6
RRGGBB
RG
RGBRGB
RGGGBB
BBBRRRRRGGGGGG
GGGGRRRB

(3) 출력

한 줄에 하나씩 몇 번의 최소 턴이 필요한지를 숫자로 출력한다.

(4) 출력 예

output.txt
4
1
5
3
8
7

=====

엘리베이터 운행거리

1. 문제

한국 엘리베이터(주)는 독신자 주거 공간에 엘리베이터를 설치해 달라는 주문을 받았다. 독신자들끼리 서로 비밀을 보장하기 위해 엘리베이터는 한 순간에 한 사람만 이용하도록 한다. 이용자가 현 위치에서 사용 버튼을 누르면 엘리베이터가 도착하고 목적층을 입력하면 이동한다. 엘리베이터가 한 층을 이동하는데 지연시간은 1초가 걸린다. 당신이 한국 엘리베이터(주)의 프로그래머라고 할 때 이 엘리베이터가 매일 움직인 거리를 측정하고자 한다.

(1) 문제 제시

한 사람만이 이용 가능한 엘리베이터를 하루에 n 명의 사람이 이용한 경우 엘리베이터가 운행한 총 층수를 구하라.

(2) 제약사항

이용자 n 명이고, n 은 $3 \leq n \leq 10$ 이며 이 건물의 엘리베이터는 -3층부터 17층까지 운행하므로 f 층수는 $-3 \leq f \leq 17$ (0은 제외)으로 입력한다.(이용자별로 출발지, 목적지를 입력하되, 같지 않아야 한다.)

시작층은 1층부터이다(문제에 없던 부분)

2. 입력 및 출력 형태

(1) 입력

첫 째 줄은 엘리베이터 이용자 수를 정수 형태로 입력하고

둘 째 줄은 이용자의 수만큼 이용자의 위치와 목적 층수를 차례로 입력한다.

(2) 입력 예

input.txt
3
-2 10
1 15
7 -1

(3)출력

엘리베이터가 처음 위치에서 마지막 이용자의 도착까지 이동한 거리(층수)를 출력한다.

(4) 출력 예

output.txt
51

=====

안전지점 찾기

1. 문제

사다리타기를 변형한 안전지점 찾기란 게임으로 커피값을 나눠서 내기로 했다. 게임의 방법은 먼저 원하는 위치(시작위치)와 횃수를 입력하면 그 위치에서부터 영문자로 동서남북을 정하고 해당 숫자만큼 이동하는 것이다. 입력된 횃수만큼을 반복하여 이동하고 최종 도착지의 숫자에 1,000을 곱한 만큼 돈을 거둔다. 범위를 벗어나는 경우 10,000원을 거둔다. 이 게임을 개발하라.

N2	S3	S2	E1	N3
W1	N1	E3	W3	S3
E3	S2	N2	S2	E1
N3	E3	S1	N1	W3
E1	N2	W1	N3	S2

[그림 1] 안전지점 찾기 게임판

맨 앞에 있는 셀의 위치를 (1,1)의 값은 N2가 된다. 그 옆에 있는 셀(S3)의 위치는 (1, 2)가 된다. 게임이 시작되고, 원하는 위치 즉, 입력된 위치가 (3, 4)이고 이동 횟수가 3이라면 (3, 4)가 S2이므로 남쪽으로 2칸 이동하여 (5, 4)에 도착한다. N3이므로 북쪽으로 3칸 이동하면 (2, 4)의 W3이므로 서쪽으로 3칸 이동하여 도착지가 W1인 (2, 1)이 된다. 숫자가 1이므로 1,000원의 회비를 내면 된다.

(1) 문제 제시

안전지점 찾기 게임을 개발하고, 게임을 시작하면서 입력한 시작위치와 이동횟수에 따라 게임을 실행했을 때 최종 도착 위치에 따른 회비를 계산하라.

(2) 제약사항

매트릭스(행렬)의 크기를 $n \times n(5 \leq n \leq 10)$ 으로 만든다.

2. 입력 및 출력 형태

(1) 입력

첫 째 줄은 테스트케이스 N을 입력한다.

둘 째 줄은 게임판의 크기 M을 입력한다.

셋 째 줄부터 $M \times M$ 크기의 2차원 배열 형태로 매트릭스(행렬)를 만들고, 그 다음 줄에 테스트 케이스만큼 시작위치(a, b)와 이동 횟수 m을 a b m 형태로 공백으로 구분하여 입력한다(단, 첨자는 1부터 시작한다.)

(2) 입력 예

input.txt
3
7
E3 S2 S4 S5 E2 S1 S5
N2 S3 S2 E1 N3 W4 N1
W1 N1 E3 W3 S3 W1 S3
E3 S2 N2 S2 E1 N2 W3
N3 E3 S1 N1 W3 E1 W1
E1 N2 W1 N3 S2 N3 W5
N5 E1 N4 W3 N5 W1 S1
3 4 3
2 3 4
4 6 5

(3) 출력

#x 형태로 시작하고 공백을 두고 그 다음에 결과를 출력한다(x는 테스트케이스 번호). 이동 중에 게임판 밖으로 넘어간 경우에는 10,000을 출력한다.

(4) 출력 예

output.txt
#1 10000
#2 2000
#3 3000

=====