
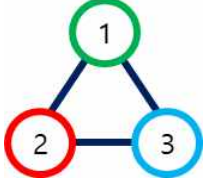
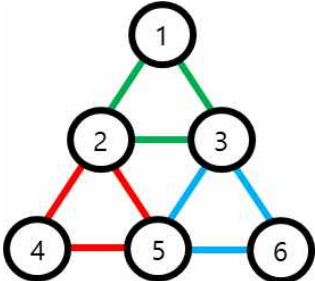


문제	
제목	낙수효과
내용	<p>금년도 동아리 엠티를 기획하는 정현이는 술자리에서 한 가지 재미있는 게임을 제안하게 되었다. 먼저 소주잔을 피라미드처럼 쌓아 올릴 것이다. 맨 위 층을 <math>L=1</math> 이라고 할 때 <math>L=1</math> 에는 잔이 1개 <math>L=2</math> 에는 잔이 3개 <math>L=3</math> 에는 잔이 6개 <math>L=k</math> 에는 <math>1+2+\dots+k</math>개를 쌓아 올린다. 소주잔의 위치는 두 개의 정수 <math>L, N</math>으로 표현된다. <math>L</math>은 소주잔의 층을 의미하며 <math>N</math>은 아래의 그림과 같이 나열되는 소주잔의 번호이다. 그림은 위에서 봤을 때 단면도이다.</p> <div style="display: flex; flex-direction: column; align-items: center;"> <div style="display: flex; align-items: center; margin-bottom: 20px;"> <div style="background-color: #92d050; padding: 5px; margin-right: 10px;"><math>L=1</math></div>  </div> <div style="display: flex; align-items: center; margin-bottom: 20px;"> <div style="background-color: #92d050; padding: 5px; margin-right: 10px;"><math>L=2</math></div>  </div> <div style="display: flex; align-items: center;"> <div style="background-color: #92d050; padding: 5px; margin-right: 10px;"><math>L=3</math></div>  </div> </div> <p>소주잔을 위와 같은 모양으로 쌓아올린 뒤에 가장 위에 있는 잔에 소주를 병으로 붓는다. (편의를 위해 소주 한 병은 7잔이라고 정한다.) 소주병을 모두 붓게 되면 잔이 가득 차게 되고 우리가 사용하는 잔은 이 게임을 위하여 특별히 제작되었기 때문에 넘치는 소주는 전혀 낭비되지 않고 이 소주잔의 밑에 있는 세 개의 소주잔에 동등하게 나눠진다. 예를 들어 <math>L=2</math> 3번 잔 밑에는 <math>\{L=3</math> 3번, <math>L=3</math> 5번, <math>L=3</math> 6번}이 위치하게 된다. 이 게임은 진행자가 정해진 <math>B</math>개의 소주병을 모두 붓고 난 뒤 특정 소주잔 <math>(L,N)</math>을 지정하면 그 소주잔이 다 채워졌는지 비어있는지 혹은 채워지다가 말았는지를 맞추는 게임</p>

		<p>이다.</p> <p>원활한 게임을 진행을 위해 정답을 미리 구하는 프로그램을 작성하라.</p> <p>사용할 수 있는 언어는 C, C++로 제한한다. 프로그램의 실행 시간은 7초 메모리 제한은 10MB이다. C++의 경우 main 함수 내의 시작 지점에 다음 내용을 추가함으로써 cin 입력 속도를 개선할 수 있다.</p> <pre>std::ios::sync_with_stdio(false);</pre>
입력 형식		<p>1. 첫 줄에는 테스트케이스의 수 <math>T(\leq 10,000)</math>가 주어진다.</p> <p>2. 각 테스트 케이스의 입력은 한 줄이며 소주병의 수 B, 소주잔의 위치 L, N이 공백으로 구분되어 주어진다.</p> <p><math>(1 \leq B \leq 20,000,000, 1 \leq L \leq 400, 1 \leq N \leq L*(L+1)/2)</math></p>
출력 형식		<p>각 테스트 케이스마다 B개의 소주병을 붓고나서 L,N의 위치에 있는 소주잔이 비어있으면 0 가득 차있으면 1 채워지다가 말았으면 2를 출력한다.</p>
예	입력	5 1 3 2 1 3 1 2 5 1 2 3 4 3 5 3 (empty line)
	출력	2 2 0 1 2 (empty line)