

1차원 수직선 위의 좌표 0 인 곳에서 시작하여 오른쪽으로 움직이면서 좌표  $x(i)$ 인 목표점에 도달하려고 한다. 움직이는 방법은 오른쪽으로 1칸, 2칸, 3칸, 등등으로 1칸씩 늘어나는 거리를 점프하는 것이다. 중간에 아무 때나 멈추고 다시 1칸으로 점프 거리를 “리셋”할 수 있다.  $f(x)$  를 위와 같은 방법으로 좌표  $x$  인 목표점에 도달하는 최소 점프 횟수라고 하자.

예를 들어,  $x=14$  라고 하면  $0 \rightarrow 1 \rightarrow 3 \rightarrow 6 \rightarrow 10(\text{리셋}) \rightarrow 11 \rightarrow 13(\text{리셋}) \rightarrow 14$  와 같이 7 번 점프하는 것이 최소 점프 횟수로  $x$  에 도달하는 방법이다.

입력으로 음이아닌 정수  $x, y$  를 받아서  $f(x), f(x+1), \dots, f(y)$

중 최대값을 계산하는 프로그램을 작성하라.

- 제한시간: 전체 테스트 케이스는 40개 이하이며, 전체 수행 시간은 2초 이내. (Java 4초 이내)  
제한 시간을 초과하면 제출한 소스코드의 프로그램이 즉시 종료되며, 그때까지 수행한 결과에서 테스트 케이스를 1개 그룹 이상 통과하였더라도 점수는 0점이 됩니다. 그러나, 제한 시간을 초과하더라도 테스트 케이스를 1개 그룹 이상 통과하였다면 '부분 점수(0; 점수; 만점)'를 받을 수 있으며, 이를 위해서는, C / C++ 에서 "printf 함수" 사용할 경우, 프로그램 시작부분에서 "setbuf(stdout, NULL);"를 한번만 사용하십시오. C++에서는 "setbuf(stdout, NULL);"와 "printf 함수" 대신 "cout"를 사용하고, Java에서는 "System.out.println"을 사용하시면, 제한 시간을 초과하더라도 '부분 점수'를 받을 수 있습니다.

※ 언어별 기본 제공 소스코드 내용 참고

만약, 제한 시간을 초과하지 않았는데도 '부분 점수'를 받았다면, 일부 테스트 케이스를 통과하지 못한 경우 입니다.

- 메모리 사용 제한 : heap, global, static 총계 256MB, stack 100MB

메모리 사용 제한

heap, global, static (총계) : 256MB

stack : 100MB

입력

입력 파일에는 여러 테스트 케이스가 포함될 수 있다.

파일의 첫째 줄에 테스트 케이스의 개수를 나타내는 자연수  $T$  가 주어지고, 이후 차례로  $T$  개의 테스트 케이스가 주어진다. ( $1 \leq T \leq 40$ ) 각 테스트 케이스의 첫 줄에는 두 정수  $x, y$ 가 주어진다. ( $0 \leq x \leq y \leq 1011$ )

- 점수 : 각 제출에서 취득한 점수 중에서 최대 점수 (만점 150점)

주어지는 테스트 케이스 데이터들의 그룹은 아래와 같으며, 각 그룹의 테스트 케이스를 모두 맞추었을 때 해당되는 부분 점수를 받을 수 있다.

• 그룹 1 (29점) : 이 그룹의 테스트 케이스에서는  $0 \leq x \leq y \leq 104$  .

• 그룹 2 (31점) : 이 그룹의 테스트 케이스에서는  $y \leq 104$ .

• 그룹 3 (90점) : 이 그룹의 테스트 케이스에서는 원래의 조건 외에는 다른 제약조건이 없다.

출력

각 테스트 케이스의 답을 순서대로 표준출력으로 출력하여야 하며,

각 테스트 케이스마다 첫 줄에는 "Case #C"를 출력하여야 한다. 이때 C는 테스트 케이스의 번호이다.

그 다음 줄에,  $f(x)$ ,  $f(x+1)$  ,  $\cdots$  ,  $f(y)$ 중 최대값을 출력한다.

입출력예

입력

2

11 16

23 28

출력

Case #1

7

Case #2

10