

문제 : Coin Move Game

문제 설명 :

아래 그림에서 보인 것처럼 정수가 차례로 적힌 1차원 배열이 있다. 번호가 양수이면서 P 의 배수인 곳과 음수인 곳에는 \times 표시가 되어있다. 번호가 S 인 칸에 동전이 하나 놓여 있고, 영희와 철수는 번갈아 가면서 그 동전을 좌측으로 이동하는 게임을 한다. 영희나 철수 모두 게임을 이기기 위한 전략을 충분히 이해하고 있다. 게임의 이동규칙과 승패 규칙은 다음과 같다.

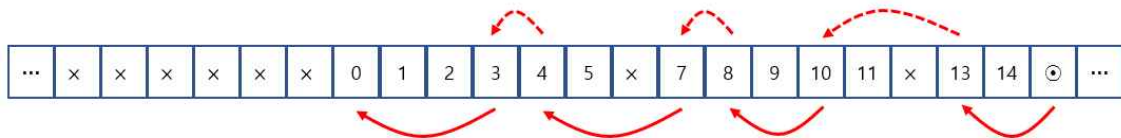
1. 게임의 시작은 영희가 한다. 즉, 영희가 동전을 먼저 이동한다.
2. 동전이 한 번에 이동할 수 있는 거리는 1이상 K 이하이다. 즉, 반드시 한 칸 이상 이동해야 하며, 최대로 이동할 수 있는 거리는 K 이다.
3. 이전 사람이 이동한 거리와 같은 거리로 동전을 이동시킬 수 없다.
4. \times 표시가 되어 있는 곳에 동전을 놓는 사람이 게임에 진다. 단, 게임 시작 시 동전의 위치에 \times 표시가 되어 있는 경우 영희는 동전을 이동할 수 있다.

예를 들어, 아래 <그림 1>은 $P=6$ 이고 (즉, 0 보다 큰 모든 P 의 배수 자리엔 \times 표시가 되어 있음), \odot 로 표시된 동전이 15번에 놓인 경우이다. 즉, $S=15$ 이다.

...	\times	\times	\times	\times	\times	\times	0	1	2	3	4	5	\times	7	8	9	10	11	\times	13	14	\odot	...
-----	----------	----------	----------	----------	----------	----------	---	---	---	---	---	---	----------	---	---	---	----	----	----------	----	----	---------	-----

<그림 1>

$K=3$ 인 경우, <그림 2>에서 보인 것처럼 영희(실선 화살표)와 철수(점선 화살표)가 동전을 이동하였다면 철수는 \times 표시가 되어 있는 곳에 동전을 이동할 수밖에 없어 게임에 지게 된다. 참고로, 영희가 첫 이동에서 동전을 13번 칸에 두면 철수가 동전을 어떻게 이동하던 영희가 게임에 이길 수 있는 방법이 있다.



P , S , K 값이 주어질 때, 영희가 게임을 이길 수 있는지, 이길 수 있다면 어디로 동전을 움직이는 것이 좋은지 그 위치를 찾는 프로그램을 작성하고자 한다.

【입 력】

입력파일의 이름은 coinmove.inp 이다. 첫째 줄에는 검사하고자 하는 총 경우의 수 T 가 주어진다. 각 테스트케이스의 첫째 줄엔 세 정수 $P(2 \leq P \leq 10)$, $K(3 \leq K \leq 7)$, $S(5 \leq S \leq 10^6)$ 가 차례로 주어진다.

【출 력】

출력 파일의 이름은 coinmove.out이다. 각 검사하는 경우에 게임을 시작하는 영희가 이길 수 있다면, 첫 이동시 동전을 어디에 놓아야 하는 지 그 번호를 출력하라. 만약 그런 위치가 여럿 있다면 가장 큰 번호를 출력하라. 만약, 영희가 이길 수 없다면 -1을 출력하라.

【실행 예】

입력 예	입력 예에 대한 출력
2	97
10 7 100	997
6 7 1000	

제한조건: 프로그램은 coinmove.{c,cpp,java}로 한다.