WC '01 Suicidal P6 - Blind Date

Time limit: 1.0s   **Memory limit:** 32M

Woburn Challenge 2001 - Suicidal

Ah, it's the end of frosh week and love is in the air. New couples are forming everywhere and everyone is happy. Hi, I'm Roger Lodge. Let's meet our couple, shall we? Simon is a Computer Engineering frosh who has a "fetish" for precision. His favourite books are "Advanced Number theory" and "The Hermit, The Hospital and The Place: baby names for mathematicians". Simone is an electrical engineering frosh who likes sparks to fly on the first date. She likes to mix psychology with physics and is currently reading "Jung on Young" and "Freud and the solenoid". Yikes.

Simon and Simone are at dinner. When their drinks (i.e. beer) arrive, it is in a large glass that holds $2N$ litres (the glass is full). In addition, the waiter has been kind enough to supply 2 other glasses: a small one that holds $N - K$ litres and a medium one holding $N + K$ litres, where $0 < K < N$.

These 2 glasses are empty. Simon, being very precise, wants to pour exactly the same amount of beer for him and his date (so that there are exactly N litres left in the large and medium sized glasses). However, the only measuring devices he has are the 3 glasses.

Since the jugs are not calibrated, the only way that he can know how much volume he is pouring is by emptying a jug fully or filling another jug fully. Dr. Love says that Simon hasn't been to the gym in a while (he's been too busy reading early 21st century literature), and so he finds the jugs heavy and wants to minimize lifting a glass and pouring it. Plus, he doesn't want to go pouring back and forth like an idiot, if it isn't even possible to split the beer evenly.

Input Specification

$N \ K$ ($0 < K < N < 4\,000$; `-1 -1` denotes end of input)

Bonus

Josh, another Waterloo keener, has solved this program for $0 < K < N < 64\,000$. Simon doesn't want to lose his reputation as the class nerd, so he's desperate to repeat this feat. If you can help him, you'll get twice as many points.

Output Specification

P ($P =$ the minimum number of pours will fit within a longint; output `infinity` if it isn't possible to split the beer evenly)

Sample Input

```
2 1
3 1
-1 -1
```

Sample Output

```
3
infinity
```