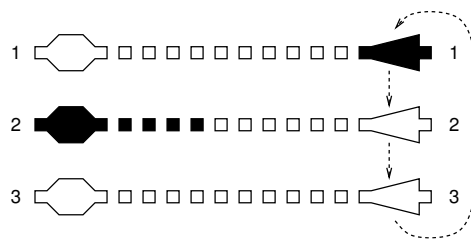


# Game-10

Nome do arquivo: `game10.c`, `game10.cpp`, `game10.pas`, `game10.java`, `game10.js` ou `game10.py`

No princípio dos anos 1980 surgiram nos colégios os primeiros relógios de pulso digitais com joguinhos. Era uma febre entre os alunos e quem tinha um era muito popular na hora do recreio. Os joguinhos eram bem simples, mas muito legais. Um dos primeiros era o Game-10, no qual você controlava um avião que aparecia na parte direita do visor. Na parte esquerda aparecia um disco voador em qualquer uma de três posições, aleatoriamente, e lançava um míssil. O objetivo do jogador era movimentar o avião verticalmente para que ficasse na frente do disco voador (na mesma linha horizontal, do lado direito) e atirar para interceptar o míssil antes que esse atingisse o avião.



Como o movimento do avião era feito com apenas um botão, só dava para movimentar em um sentido: ao apertar o botão sucessivas vezes, o avião se movia na sequência de posições  $\dots 1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \dots$ . Veja que, na situação da figura, o jogador deveria apertar o botão apenas uma vez, para ir da posição 1 para a posição 2, e conseguir atirar e interceptar o míssil.

Neste problema vamos considerar que existem  $N$  posições e não apenas três. Dado o número de posições  $N$ , a posição  $D$  na qual o disco voador aparece, e a posição  $A$  onde está o avião, seu programa deve computar o número mínimo de vezes que o jogador precisa apertar o botão para movimentar o avião até a mesma posição do disco voador e poder atirar!

## Entrada

A primeira linha da entrada contém um inteiro  $N$ , o número de posições. A segunda linha contém um inteiro  $D$ , a posição do disco voador. A terceira linha contém um inteiro  $A$ , a posição do avião.

## Saída

Seu programa deve imprimir uma linha contendo um inteiro, o número mínimo de vezes que o jogador deve apertar o botão para poder atirar.

## Restrições

- $3 \leq N \leq 100$
- $1 \leq D, A \leq N$

## Exemplos

Entrada	Saída
3 2 1	1
Entrada	Saída
20 8 13	15