



STUDENT REPORT

DETAILS

Name

Sravani

Roll Number

382388097

EXPERIMENT

Title

CANDIES

Description

Let's consider a scenario where there are K candies to be distributed among N children, each uniquely numbered from 1 to N . The distribution commences with Child A , followed by a sequential allocation to the subsequent children in the order: $A, A+1, A+2, \dots, N$. The query at hand is to identify which child will be the last recipient of a candy.

In more explicit terms, after Child x (where $1 \leq x < N$) receives a candy, the subsequent candy is granted to Child $x+1$. Upon Child N receiving a candy, the distribution cycle restarts, and Child 1 becomes the next recipient.

The primary objective is to ascertain the identity of the child who will receive the last candy in this cyclic distribution.

Note: Each child receives only 1 candy.

Input Format:

The first line of input contains 3 space separated integers N, K and A .

Output Format:

Print the friend who will be the final recipient of the candy.

Constraints:

$1 \leq N \leq K \leq 10^8$

Sample Input:

5 2 1

Sample Output:

2

Source Code:

```

def two_sum(nums, target):
    num_to_index = {} # Dictionary to hold number and its index

    for index, num in enumerate(nums):
        complement = target - num # Calculate the complement

        # Check if the complement is in the dictionary
        if complement in num_to_index:
            return [num_to_index[complement], index] # Return the indices

        # Store the number and its index in the dictionary
        num_to_index[num] = index

# Example usage
if __name__ == "__main__":
    import sys

    nums = list(map(int, sys.stdin.readline().strip().split())) # Read the list of integers
    target = int(sys.stdin.readline().strip()) # Read the target sum

    result = two_sum(nums, target)
    print(result)

```

RESULT

0 / 6 Test Cases Passed | 0 %