# Fibonacci Modified

**Problem Statement**

A series is defined in the following manner:

Given the $n^{th}$ and $(n+1)^{th}$ terms, the $(n+2)^{th}$ can be computed by the following relation
$\mathbf{T_{n+2} = (T_{n+1})^2 + T_n}$

So, if the first two terms of the series are 0 and 1:
the third term = $1^2 + 0 = 1$
fourth term = $1^2 + 1 = 2$
fifth term = $2^2 + 1 = 5$
... And so on.

Given three integers **A**, **B** and **N**, such that the first two terms of the series ($1^{st}$ and $2^{nd}$ terms) are **A** and **B** respectively, compute the $N^{th}$ term of the series.

**Input Format**

You are given three space separated integers **A, B and N** on one line.

**Input Constraints**
0 <= **A,B** <= 2
3 <= **N** <= 20

**Output Format**

One integer.
This integer is the $\mathbf{N^{th}}$ term of the given series when the first two terms are **A** and **B** respectively.

**Note**

- Some output may even exceed the range of 64 bit integer.

**Sample Input**

```
0 1 5
```

**Sample Output**

```
5
```

**Explanation**

The first two terms of the series are 0 and 1. The fifth term is 5. How we arrive at the fifth term, is explained step by step in the introductory sections.