# Integers Come In All Sizes

Problem Submissions

Integers in Python can be as big as the bytes in your machine's memory. There is no limit in size as there is: $2^{31} - 1$ (c++ int) or $2^{63} - 1$ (C++ long long int).

As we know, the result of $a^b$ grows really fast with increasing $b$.

Let's do some calculations on very large integers.

**Task**
Read four numbers, $a$, $b$, $c$, and $d$, and print the result of $a^b + c^d$.

**Input Format**
Integers $a$, $b$, $c$, and $d$ are given on four separate lines, respectively.

**Constraints**
$1 \leq a \leq 1000$
$1 \leq b \leq 1000$
$1 \leq c \leq 1000$
$1 \leq d \leq 1000$

**Output Format**
Print the result of $a^b + c^d$ on one line.

**Sample Input**

```
9
29
7
27
```

**Sample Output**

```
4710194409608608369201743232
```

**Note**: This result is bigger than $2^{63} - 1$. Hence, it won't fit in the long long int of C++ or a 64-bit integer.



Current Buffer (saved locally, editable) Python 3

```
a=int(input())
b=int(input())
c=int(input())
d=int(input())
print(a ** b + c ** d)
```

Line: 5 Col: 23

Upload Code as File  Test against custom input  Run Code  Submit Code