The Power Sum

Find the number of ways that a given integer, X, can be expressed as the sum of the N^{th} powers of unique, natural numbers.

For example, if $X = 13$ and $N = 2$, we have to find all combinations of unique squares adding up to 13. The only solution is $2^2 + 3^2$.

Function Description

Complete the *powerSum* function in the editor below. It should return an integer that represents the number of possible combinations.

powerSum has the following parameter(s):

- *X*: the integer to sum to
- *N*: the integer power to raise numbers to

Input Format

The first line contains an integer X.
The second line contains an integer N.

Constraints

- $1 \leq X \leq 1000$
- $2 \leq N \leq 10$

Output Format

Output a single integer, the number of possible combinations caclulated.

Sample Input 0

```
10
2
```

Sample Output 0

```
1
```

Explanation 0

If $X = 10$ and $N = 2$, we need to find the number of ways that 10 can be represented as the sum of squares of unique numbers.

$10 = 1^2 + 3^2$

This is the only way in which 10 can be expressed as the sum of unique squares.

Sample Input 1

```
100
2
```

Sample Output 1

```
3
```

Explanation 1

$100 = (10^2) = (6^2 + 8^2) = (1^2 + 3^2 + 4^2 + 5^2 + 7^2)$

Sample Input 2

```
100
3
```

Sample Output 2

```
1
```

Explanation 2

100 can be expressed as the sum of the cubes of $1, 2, 3, 4$.
$(1 + 8 + 27 + 64 = 100)$. There is no other way to express 100 as the sum of cubes.

Contest ends in 10 hours

Submissions: 51
Max Score: 50
Difficulty: Medium

Rate This Challenge:

More

C++

```
#include <bits/stdc++.h>

using namespace std;

string ltrim(const string &);
string rtrim(const string &);

/*
 * Complete the 'powerSum' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts following parameters:
 *  1. INTEGER X
 *  2. INTEGER N
 */

int powerSum(int X, int N) {
```

```
18
19  }
20
21  int main()
22  {
23      ofstream fout(getenv("OUTPUT_PATH"));
24
25      string X_temp;
26      getline(cin, X_temp);
27
28      int X = stoi(ltrim(rtrim(X_temp)));
29
30      string N_temp;
31      getline(cin, N_temp);
32
33      int N = stoi(ltrim(rtrim(N_temp)));
34
35      int result = powerSum(X, N);
36
37      fout << result << "\n";
38
39      fout.close();
40
41      return 0;
42  }
43
44  string ltrim(const string &str) {
45      string s(str);
46
47      s.erase(
48          s.begin(),
49          find_if(s.begin(), s.end(), not1(ptr_fun<int, int>(isspace)))
50      );
51
52      return s;
53  }
54
55  string rtrim(const string &str) {
56      string s(str);
57
58      s.erase(
59          find_if(s.rbegin(), s.rend(), not1(ptr_fun<int, int>(isspace))).base(),
60          s.end()
61      );
62
63      return s;
64  }
65
```

Line: 1 Col: 1

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