

GE23131-Programming Using C-2024

Quiz navigation



Show one page at a time

Finish review

Status	Finished
Started	Sunday, 12 January 2025, 9:53 PM
Completed	Sunday, 12 January 2025, 10:49 PM
Duration	55 mins 14 secs

Question **1**

Correct

Marked out of 1.00

 [Flag question](#)

You are a bank account hacker. Initially you have 1 rupee in your account, and you want exactly **N** rupees in your account. You wrote two hacks, first hack can multiply the amount of money you own by 10, while the second can multiply it by 20. These hacks can be used any number of time. Can you achieve the desired amount **N** using these hacks.

Constraints:

$1 \leq T \leq 100$

$1 \leq N \leq 10^{12}$

Input

- The test case contains a single integer N.

Output



SAMPLE INPUT

1

SAMPLE OUTPUT

1

SAMPLE INPUT

2

SAMPLE OUTPUT

0

Answer: (penalty regime: 0 %)

Reset answer

```
1 1 /*
2   * Complete the 'myFunc' function below.
3   *
4   * The function is expected to return an INTEGER.
5   * The function accepts INTEGER n as parameter.
6   */
7
8 2 int myFunc(int n) {
9   // If n is less than 1, it's impossible to achieve
```


```
12     }
13
14     // Working backwards from n to 1 by dividing by 10 or 20
15     while (n > 1) {
16         // If divisible by 20, divide by 20
17         if (n % 20 == 0) {
18             n /= 20;
19         }
20         // If divisible by 10, divide by 10
21         else if (n % 10 == 0) {
22             n /= 10;
23         }
24         // If not divisible by either 10 or 20, it's impossible
25         else {
26             return 0;
27         }
28     }
29
30     // If we reached exactly 1, it's possible
31     return (n == 1);
32 }
```

	Test	Expected	Got	
✓	printf("%d", myFunc(1))	1	1	✓
✓	printf("%d", myFunc(2))	0	0	✓
✓	printf("%d", myFunc(10))	1	1	✓
✓	printf("%d", myFunc(25))	0	0	✓
✓	printf("%d", myFunc(200))	1	1	✓

Passed all tests! ✓



Marked out of
1.00

 [Flag question](#)

N 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

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



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.

Answer: (penalty regime: 0 %)

Reset answer

```
1 | /*
2 |  * Complete the 'powerSum' function below.
3 |  *
```



```
7  * 2. INTEGER n
8  */
9
10 #include <stdio.h>
11 #include <math.h>
12
13 int powerSum(int x, int m, int n) {
14     if (x == 0)
15         return 1;
16     if (x < 0 || pow(m, n) > x)
17         return 0;
18
19     return powerSum(x - pow(m, n), m + 1, n) + powerSum(x, m + 1, n);
20 }
```

	Test	Expected	Got	
✓	printf("%d", powerSum(10, 1, 2))	1	1	✓

Passed all tests! ✓

Finish review

