Started	Monday, 13 January 2025, 1:20 PM
Completed	Monday, 13 January 2025, 1:31 PM
Duration	10 mins 41 secs
Correct	A binary number is a combination of 1s and 0s. Its n th least significant digit is the n th digit starting from the right starting with 1. Given a decimal number, convert it to binary and determine the value of the the 4 th least significant digit.
♥ Flag question	Example
	number = 23
	Convert the decimal number 23 to binary number: $23^{10} = 2^4 + 2^2 + 2^1 + 2^0 = (10111)_2$.
	The value of the 4 th index from the right in the binary representation is 0.
	Function Description
•	Complete the function fourthBit in the editor below.

Status Finished

```
fourthBit has the following parameter(s):
int number: a decimal integer
Returns:
int: an integer 0 or 1 matching the 4th least significant digit in the binary representation of number.
Constraints
0 \le \text{number} < 2^{31}
Input Format for Custom Testing
Input from stdin will be processed as follows and passed to the function.
The only line contains an integer, number.
Sample Case 0
```

Sample Input 0

```
STDIN Function
32 \rightarrow \text{number} = 32
Sample Output 0
0
Explanation 0
      Convert the decimal number 32 to binary number: 32_{10} = (100000)_2.
      The value of the 4th index from the right in the binary representation is 0.
Sample Case 1
Sample Input 1
STDIN Function
77 \rightarrow \text{number} = 77
```

Sample Output 1

1

Explanation 1

- Convert the decimal number 77 to binary number: $77_{10} = (1001101)_2$.
- The value of the 4th index from the right in the binary representation is 1.

Answer: (penalty regime: 0 %)

Reset answer

	Test	Expected	Got	
~	<pre>printf("%d", fourthBit(32))</pre>	0	0	~
~	<pre>printf("%d", fourthBit(77))</pre>	1	1	~

Passed all tests! <

Question 2 Correct Marked out of 1.00	Determine the factors of a number (i.e., all positive integer values that evenly divide into a number) and then return the p^{th} element of the list, sorted ascending. If there is no p^{th} element, return 0.	
⟨ Flag question	Example	
	n = 20 p = 3	
	The factors of 20 in ascending order are $\{1, 2, 4, 5, 10, 20\}$. Using 1-based indexing, if $p = 3$, then 4 is returned. If $p > 6$, 0 would be returned.	
	Function Description	
	Complete the function pthFactor in the editor below.	
	pthFactor has the following parameter(s):	
	int n: the integer whose factors are to be found	
	int p: the index of the factor to be returned	

Returns: int: the long integer value of the pth integer factor of n or, if there is no factor at that index, then 0 is returned Constraints $1 \le n \le 10^{15}$ $1 \le p \le 10^9$ Input Format for Custom Testing Input from stdin will be processed as follows and passed to the function. The first line contains an integer n, the number to factor. The second line contains an integer p, the 1-based index of the factor to return. Sample Case 0 Sample Input 0 STDIN Function

Explanation 0

Factoring n = 10 results in $\{1, 2, 5, 10\}$. Return the $p = 3^{rd}$ factor, 5, as the answer.

Sample Case 1

Sample Input 1

STDIN Function

$$10 \rightarrow n = 10$$

$$5 \rightarrow p = 5$$

Sample Output 1 0 **Explanation 1** Factoring n = 10 results in $\{1, 2, 5, 10\}$. There are only 4 factors and p = 5, therefore 0 is returned as the answer. Sample Case 2 Sample Input 2 STDIN Function $1 \rightarrow n = 1$

 $1 \rightarrow p = 1$

Sample Output 2

Explanation 2

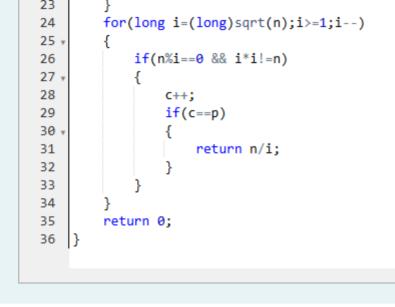
Factoring n = 1 results in {1}. The p = 1st factor of 1 is returned as the answer.

Answer: (penalty regime: 0 %)

Reset answer

22

```
1 • /*
     * Complete the 'pthFactor' function below.
 3
     * The function is expected to return a LONG INTEGER.
4
     * The function accepts following parameters:
     * 1. LONG INTEGER n
        2. LONG INTEGER p
 7
    #include<math.h>
     long pthFactor(long n, long p)
10
11 *
         int c=0;
12
13
         for(long i=1;i*i<=n;i++)</pre>
14 *
15
            if(n\%i==0)
16 •
17
                C++;
18
                if(c==p)
19 •
20
                     return i;
21
```



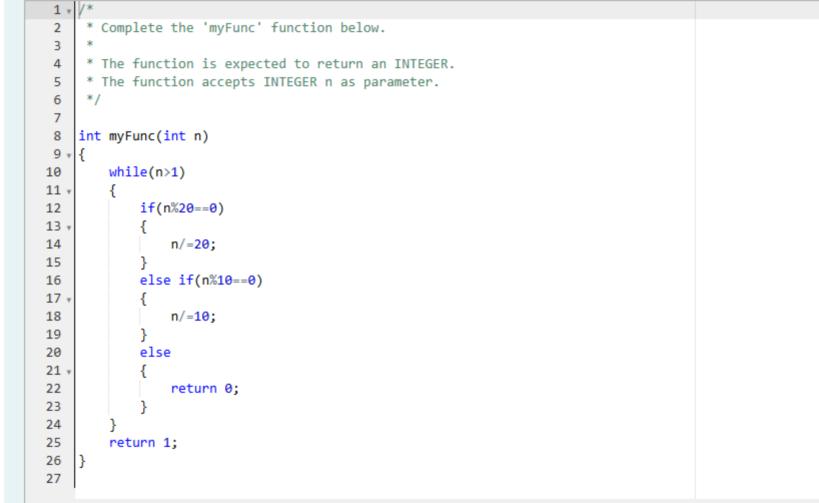
Passed all tests! <

	Test	Expected	Got	
~	printf("%ld", pthFactor(10, 3))	5	5	~
~	printf("%ld", pthFactor(10, 5))	0	0	~
~	printf("%ld", pthFactor(1, 1))	1	1	~

Status	Finished
Started	Monday, 13 January 2025, 1:32 PM
Completed	Monday, 13 January 2025, 2:06 PM
Duration	33 mins 43 secs
Correct Marked out of	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<=T<=100
	1<=N<=10^12
	Input
	· The test case contains a single integer N.
	Output

For each test case, print a single line containing the string "1" if you can make exactly N rupees or "0" otherwise.
SAMPLE INPUT
1
SAMPLE OUTPUT
1
SAMPLE INPUT
2
SAMPLE OUTPUT
0
Answer: (penalty regime: 0 %)

Reset answer 4



	Test	Expected	Got	
~	<pre>printf("%d", myFunc(1))</pre>	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! ✓

Qu	estion 2
Со	rrect
Ma 1.0	arked out of 0
P	Flag question

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 2² + 3².

Function Description

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

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

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 $1 \le X \le 1000$ $2 \le N \le 10$ **Output Format** Output a single integer, the number of possible combinations calculated. Sample Input 0 10 2 Sample Output 0 **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

Sample Output 1

Explanation 1

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

$$4^2 + 5^2 + 7^2$$

Sample Output 2 **Explanation 2** Answer: (penalty regime: 0 %)

Reset answer

Sample Input 2

100

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.

```
1 | /*
 2
     * Complete the 'powerSum' function below.
     * The function is expected to return an INTEGER.
 4
 5
     * The function accepts following parameters:
        1. INTEGER x
        2. INTEGER n
    #include<math.h>
10
    int powerSum(int x, int m, int n)
11 v {
12
        int t=pow(m,n);
13
        if(t==x)
14 *
15
            return 1;
16
17
        else if(t>x)
18 ,
19
            return 0;
20
        else
21
22 v
23
            return powerSum(x,m+1,n)+powerSum(x-t,m+1,n);
24
25
```

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

Passed all tests! ✓