



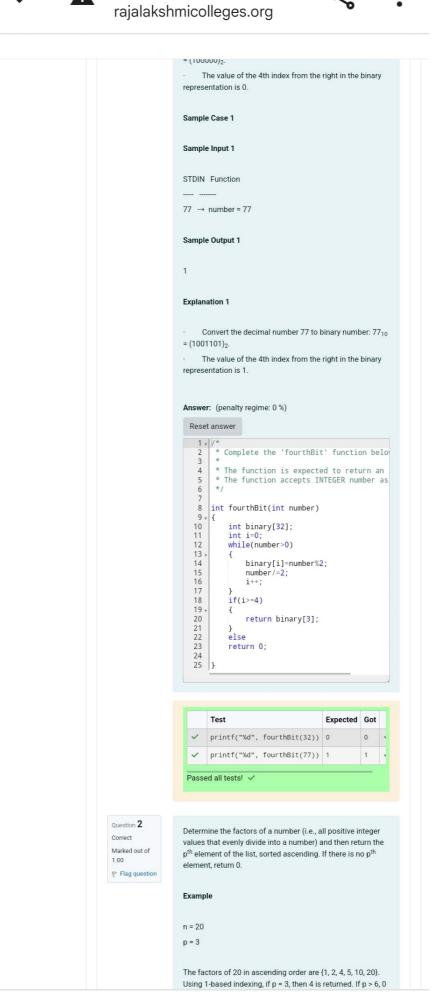


Coding: Attempt rev...



:

REC-CIS



Sample Case 1

Sample Input 1

Sample Output 1

n

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 → n = 1

1 → p = 1
```

Sample Output 2

1

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
        * Complete the 'pthFactor' function below
       * The function is expected to return a Li
* The function accepts following paramet

* 1. LONG_INTEGER n

* 2. LONG_INTEGER p
  6
 10 long pthFactor(long n, long p)
           int count=0;
for(long i=1;i<=n;++i)</pre>
 12
 13
 14 +
 15
                 if(n%i==0)
 16 +
 17
                       count++;
 18
                       if(count==p)
 19 +
 20
                            return i;
 21
 22
 23
24
            return 0;
 25
```

```
Test Expected G

v printf("%ld", pthFactor(10, 3)) 5 5

v printf("%ld", pthFactor(10, 5)) 0 0

v printf("%ld", pthFactor(1, 1)) 1 1

Passed all tests! v
```

```
I II you call make exactly is tupees of o otherwise
SAMPLE INPUT
SAMPLE OUTPUT
SAMPLE INPUT
SAMPLE OUTPUT
Answer: (penalty regime: 0 %)
 Reset answer
        * Complete the 'myFunc' function below.
   2
   4
   5
   6
   8 int myFunc(int n)
   9 + {
  10
           while(n>1)
```

```
* The function is expected to return an
* The function accepts INTEGER n as para
11 +
              if(n%20==0)
12
13 +
                  n/=20;
14
15
              else if(n%10==0)
16
17 *
                  n/=10;
18
19
20
              else
21 +
22
                  return 0;
23
24
25
         return 1;
26
27 }
28
```

✓ printf	("%d", myFunc(1)) ("%d", myFunc(2))	0	1	~
	("%d", myFunc(2))	0		
✓ printf			0	~
F (570070)	("%d", myFunc(10))	1	1	~
✓ printf	("%d", myFunc(25))	0	0	~
✓ printf	("%d", myFunc(200))	1	1	~

Question 2

Correct

Marked out of 1.00

1.00 P Flag question

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

```
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.
Answer: (penalty regime: 0 %)
 Reset answer
         * Complete the 'powerSum' function below
    3
        \boldsymbol{\ast} The function is expected to return an
    4
        * The function accepts following paramet

* 1. INTEGER x

* 2. INTEGER n

*/
    5 6 7 8
   10
        int powerSum(int x, int m, int n)
   11 +
   12
             if(x==0)
   13
                 return 1;
   14
   15
            if(x<0)
   16
   17
                 return 0;
   18
   19
   20
            int count =0;
   21
             for(int i=m;;i++)
   22
                 int power=1;
for(int j=0;j<n;j++)</pre>
   23
   24
   25
                      power*=i;
  26
27
                 if(power>x)
   28
   29
   30
                      break;
   31
                 count+=powerSum(x-power,i+1,n);
   32
   33
  34
35
             return count;
   36 }
```

```
Test Expected (

right printf("%d", powerSum(10, 1, 2)) 1

Passed all tests!
```