

```
1 | /*
     * Complete the 'fourthBit' function below.
 3
 4
     * The function is expected to return an INTEGER.
     * The function accepts INTEGER number as parameter.
 5
 6
7
    int fourthBit(int number)
 8
9 ▼ {
10
        int binary[32];
11
        int i=0;
        while(number>0)
12
13 🔻
14
            binary[i]= number%2;
            number/=2;
15
16
            i++;
17
        if(i>=4)
18
19 ▼
        {
20
            return binary[3];
21
22
        else return 0;
23 }
```

	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

Flag question

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.

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

```
10
    long pthFactor(long n, long p)
11
12
        int count = 0;
        for (long i = 1; i <= n; ++i)
13
14
15
             if(n\%i==0)
16
                 count++;
17
18
                 if(count ==p)
19
                     return i;
20
21
22
23
24
        return 0;
25 }
```

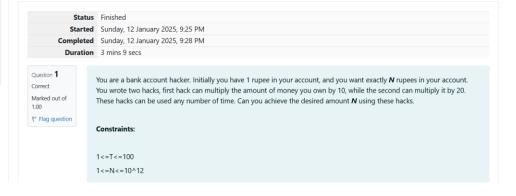
	Test	Expected	Got	
~	<pre>printf("%ld", pthFactor(10, 3))</pre>	5	5	~
~	<pre>printf("%ld", pthFactor(10, 5))</pre>	0	0	~
~	<pre>printf("%ld", pthFactor(1, 1))</pre>	1	1	~

Passed all tests! <

Quiz navigation

1 2

Show one page at a time



```
2
     * Complete the 'myFunc' function below.
3
4
     * The function is expected to return an INTEGER.
     \ensuremath{^{*}} The function accepts INTEGER n as parameter.
5
6
8
    int myFunc(int n)
9
    {
         return n==1 | | n\%10==0;
10
    }
11
12
```

	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! <

Question **2**Correct
Marked out of 1.00

F Flag question

Find the number of ways that a given integer, \mathbf{X} , can be expressed as the sum of the \mathbf{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):