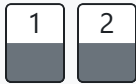


## Quiz navigation

[Show one page at a time](#)[Finish review](#)

|                  |                                   |
|------------------|-----------------------------------|
| <b>Status</b>    | Finished                          |
| <b>Started</b>   | Tuesday, 14 January 2025, 7:21 AM |
| <b>Completed</b> | Tuesday, 14 January 2025, 7:27 AM |
| <b>Duration</b>  | 6 mins                            |

Question **1**

Correct

Marked out of  
1.00[Flag question](#)

A binary number is a combination of 1s and 0s. Its  $n^{\text{th}}$  least significant digit is the  $n^{\text{th}}$  digit starting from the right starting with 1. Given a decimal number, convert it to binary and determine the value of the the 4<sup>th</sup> least significant digit.

**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<sup>th</sup> index from the right in the binary representation is 0.

**Function Description**

Complete the function fourthBit in the editor below.

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.

$$0 \leq \text{number} < 2^{31}$$

**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     $\rightarrow$  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     $\rightarrow$  number = 77

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

```
1  /*
2   * Complete the 'fourthBit'
3   *
4   * The function is expected
5   * The function accepts INTE
6   */
7
8  int fourthBit(int number)
9  {
10     int binary[32];
11     int i=0;
12     while(number>0)
13     {
14         binary[i]=number%2;
15         number/=2;
16         i++;
17     }
18     if(i>=4)
19     {
20         return binary[3];
21     }
22     else
23     return 0;
24
25 }
```

|   | Test                           | Expected | Got |
|---|--------------------------------|----------|-----|
| ✓ | printf("%d",<br>fourthBit(32)) | 0        | 0   |
| ✓ | printf("%d",<br>fourthBit(77)) | 1        | 1   |

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  $p^{\text{th}}$  element of the list, sorted ascending. If there is no  $p^{\text{th}}$  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

Returns:

`int`: the long integer value of the  $p^{\text{th}}$  integer factor of  $n$  or, if there is no factor at that index, then 0 is returned

**Constraints** $1 \leq n \leq 10^{15}$  $1 \leq p \leq 10^9$

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 |
|-------|----------|
| ----- | -----    |
| 10    | → n = 10 |
| 3     | → p = 3  |

**Sample Output 0**

5

**Explanation 0**

Factoring n = 10 results in {1, 2, 5, 10}.  
Return the p = 3<sup>rd</sup> factor, 5, as the answer.

**Sample Case 1**

**Sample Input 1**

| STDIN | Function |
|-------|----------|
| ----- | -----    |
| 10    | → n = 10 |
| 5     | → p = 5  |

**Sample Output 1**

0

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  $p = 1$ st  
factor of 1 is returned as the answer.

**Answer:** (penalty regime: 0 %)

Reset answer

```

1  /*
2  * Complete the 'pthFactor'
3  *
4  * The function is expected
5  * The function accepts foll
6  * 1. LONG_INTEGER n
7  * 2. LONG_INTEGER p
8  */
9
10 long pthFactor(long n, long
11 {
12     int count=0;
13     for(long i=1;i<=n;++i)
14     {
15         if(n%i==0)
16         {
17             count++;
18             if(count==p)
19             {
20                 return i;

```

```
23 |     }  
24 |     return 0;  
25 | }
```

|   | Test                                  | Expected | Got |
|---|---------------------------------------|----------|-----|
| ✓ | printf("%ld",<br>pthFactor(10,<br>3)) | 5        | 5   |
| ✓ | printf("%ld",<br>pthFactor(10,<br>5)) | 0        | 0   |
| ✓ | printf("%ld",<br>pthFactor(1,<br>1))  | 1        | 1   |

Passed all tests! ✓

Finish review