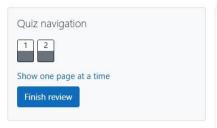
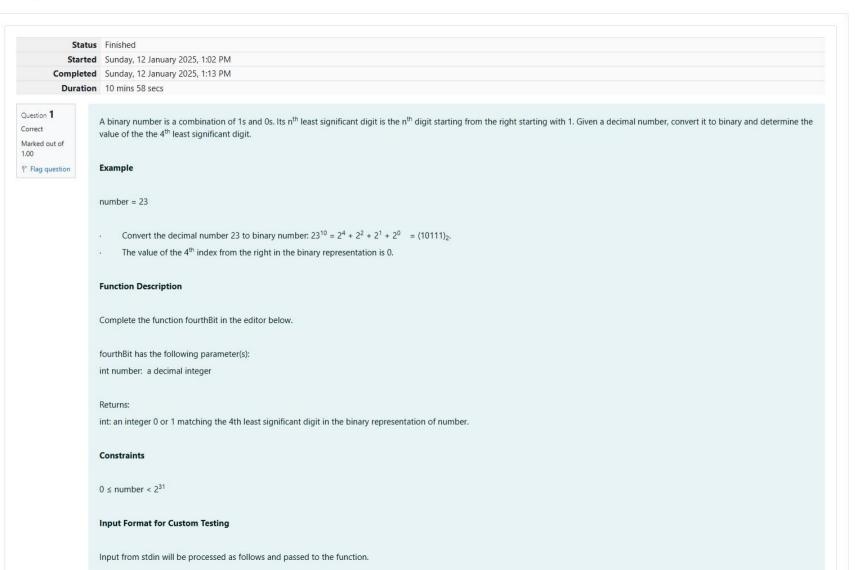
**S2** 

# **SEJALSAI S 2024-CSE**

REC-CIS

# GE23131-Programming Using C-2024





```
STDIN Function
32 → number = 32
Sample Output 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 → number = 77
Sample Output 1
Explanation 1
   Convert the decimal number 77 to binary number: 77<sub>10</sub> = (1001101)<sub>2</sub>.
     The value of the 4th index from the right in the binary representation is 1.
Answer: (penalty regime: 0 %)
 Reset answer
        * Complete the 'fourthBit' function below.
   3
        * The function is expected to return an INTEGER.
   5
        * The function accepts INTEGER number as parameter.
    6
   7
   8 int fourthBit(int number)
   9 , {
  10
           int binary[32];
```

0

1

int i=0;
while(number>0)

number/=2; i++;

binary[i]=number%2;

15

```
17
18
       if(i>=4)
19 +
          return binary[3];
20
21
22
23
       else
24
       return 0;
25 }
```

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

Passed all tests! ✓

Correct Marked out of 1.00

F Flag question

Ouestion 2

Determine the factors of a number (i.e., all positive integer values that evenly divide into a number) and then return the pth element of the list, sorted ascending. If there is no pth element, return 0.

## Example

p = 3

n = 20

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
10 \rightarrow n = 10
3 \rightarrow p = 3
```

## Sample Output 0

5

## 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 → n = 1

```
1 \rightarrow 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

```
2
      * Complete the 'pthFactor' function below.
     * The function is expected to return a LONG_INTEGER.

* The function accepts following parameters:

* 1. LONG_INTEGER n

* 2. LONG_INTEGER p
4
 5
 6
 7
10 long pthFactor(long n, long p)
11 + {
         int count=0;
12
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;
21
22
23
24
         return 0;
25 }
```

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

Passed all tests! <

Finish review