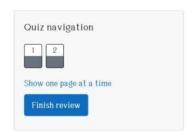
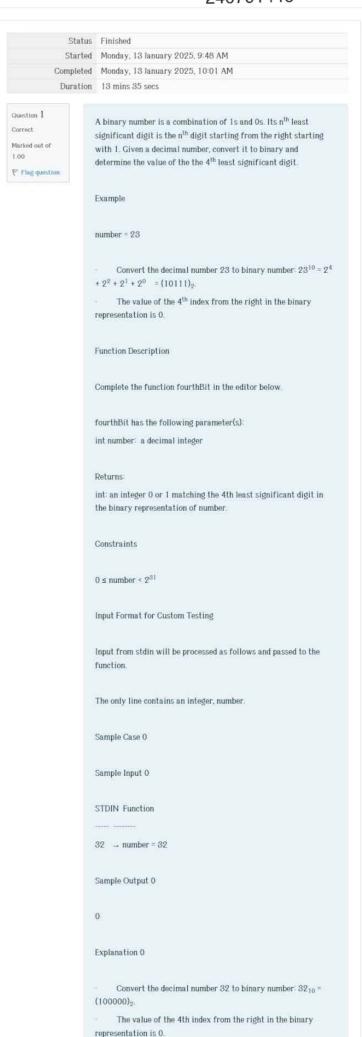
GE23131-Programming Using C-2024

Sabariprabu.M 240701446





```
Sample Case 1
Sample Input 1
STDIN Function
77 → number = 77
Sample Output 1
1
Explanation 1
    Convert the decimal number 77 to binary number: 77<sub>10</sub> =
(1001101)2-
    The value of the 4th index from the right in the binary
representation is 1.
Answer: (penalty regime: 0 %)
  Reset answer
  1.
        int fourthBit(int number)
    5
             int binary[32];
    6
             int i=0;
             while(number>0)
```

```
Test Expected Got

printf("%d", fourthBit(32)) 0 0 

printf("%d", fourthBit(77)) 1 1 

Passed all tests! ✓
```

Guestion 2
Correct
Marked out of 1.00

P 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

Returns:

```
int: the long integer value of the p<sup>th</sup> 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 → n=10
3 - p = 3
Sample Output 0
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 → n=10
5 → p=5
Sample Output 1
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 \quad \rightarrow \quad p=1
Sample Output 2
```

Explanation 2

```
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 → n = 10
5 - 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 - 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
 1 |long pthFactor(long n, long p)
    3
            int count=0;
for(long i=1;i<=n;++i)</pre>
           if(n%i==0)
{
    count++;
    if(count=
    5 +
    8
                    if(count==p)
{
   10 +
                          return i;
   12
13
14
   15
            return 0;
   16 }
       Test
                                           Expected Go

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

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

✓ printf("%ld", pthFactor(1, 1)) 1
                                                     1
 Passed all tests!
```

Finish review