REC-CIS Maanisha E - 240901055

GE23131-Programming Using C-2024

|  |  |
| --- | --- |
| **Status** | Finished |
| **Started** | Thursday, 26 December 2024, 8:13 AM |
| **Completed** | Thursday, 26 December 2024, 8:21 AM |
| **Duration** | 8 mins 27 secs |

A binary number is a combination of 1s and 0s. Its nth least significant digit is the nth digit starting from the right starting with 1. Given a decimal number, convert it to binary and determine the value of the the 4th least significant digit.



Question **1**

Correct

Marked out of 1.00

Flag question

Quiz navigation

1

2

[Show one page at a time](http://www.rajalakshmicolleges.org/moodle/mod/quiz/review.php?attempt=137286&cmid=194&showall=0)

Finish review

# Example

number = 23

* Convert the decimal number 23 to binary number: 2310 = 24 + 22 + 21 + 20 =

(10111)2.

* The value of the 4th index from the right in the binary representation is 0.

# Function Description

Complete the function fourthBit in the editor below.

REC-CIS

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.

**Constraints**

0 ≤ number < 231

**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 → number = 32

**Sample Output 0**

REC-CIS

0

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

Convert the decimal number 77 to binary number: 7710 = (1001101)2.

The value of the 4th index from the right in the binary representation is 1.

·

·

**Explanation 1**

1

**Sample Output 1**

77 → number = 77

STDIN Function

**Sample Input 1**

**Sample Case 1**

Convert the decimal number 32 to binary number: 3210 = (100000)2.

The value of the 4th index from the right in the binary representation is 0.

·

·

**Explanation 0**

REC-CIS

1 ▼

Reset answer

2

3

4

5

6

7

8

9 ▼

10

11

12

13 ▼

14

15

16

17

18

19 ▼

20

21

22

23

24



Passed all tests!

/\*

* Complete the 'fourthBit' function below.

\*

* The function is expected to return an INTEGER.
* The function accepts INTEGER number as parameter.

\*/

int fourthBit(int number)

{

int binary[32]; int i=0;

while(number>0)

{

binary[i]=number%2; number/=2;

i++;

}

if(i>=4)

{

return binary[3];

}

else

return 0;

}



Question **2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Test** | **Expected** | **Got** |  |
|  | printf("%d", fourthBit(32)) | 0 | 0 |  |
|  | printf("%d", fourthBit(77)) | 1 | 1 |  |

REC-CIS



1.00

Correct Determine the factors of a number (i.e., all positive integer values that evenly divide

Marked out of into a number) and then return the pth element of the list, sorted ascending. If there is

Flag question

no pth 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 pth integer factor of n or, if there is no factor at that index, then 0 is returned

# Constraints

1 ≤ n ≤ 1015

REC-CIS

1 ≤ p ≤ 109

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**

5

**Explanation 0**

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

REC-CIS

**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

REC-CIS

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

1 ▼

/\*

* 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

\*/

long pthFactor(long n, long p)

{

int count=0;

for(long i=1;i<=n;i++)

{

if(n%i==0)

{

count++;

if(count==p)

{

return i;

}

}

}

return 0;

}

3

4

5

6

7

8

9

10

11 ▼

12

13

14 ▼

15

16 ▼

17

18

19 ▼

20

21

22

23

24

25

|  |  |  |
| --- | --- | --- |
| REC-CIS |  |  |
| printf("%ld", pthFactor(10, 3)) | 5 | 5 |
| printf("%ld", pthFactor(10, 5)) | 0 | 0 |
| printf("%ld", pthFactor(1, 1))  Passed all tests! | 1 | 1  Finish review |

**Test Expected Got**