

Odd numbers in N-th row of Pascal's Triangle

Given N, the row number of Pascal's triangle(row starting from 0). Find the count of odd numbers in N-th row of Pascal's Triangle.

[Hire with us!](#)

Prerequisite : [Pascal's Triangle](#) | [Count number of 1's in binary representation of N](#)

Examples :

广告 X

Enabling HLS for FPGA

Leveraging standard HLS tools from FPGA

Silexica

OPEN

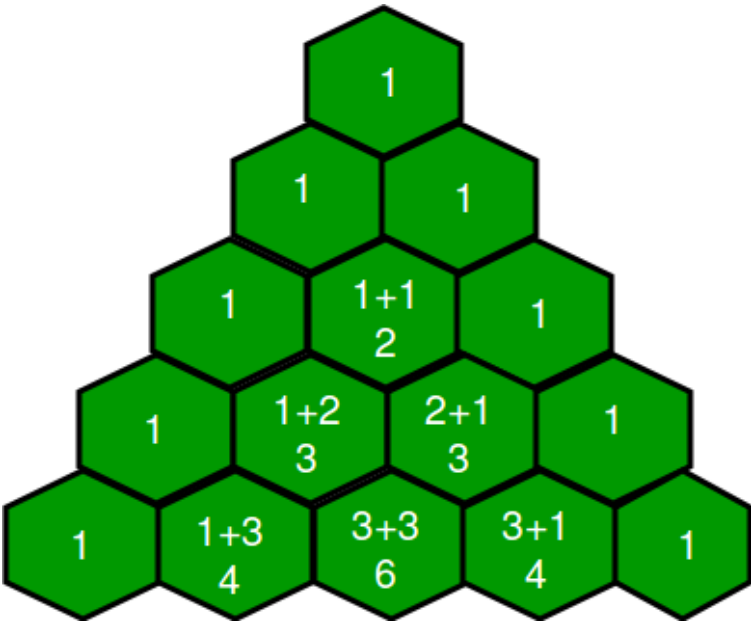
Input : 11

Output : 8

--->

Input : 20

Output : 4



Approach : It appears the answer is always a power of 2. In fact, the following theorem exists :

THEOREM : The number of odd entries in row N of Pascal's Triangle is 2 raised to the number of 1's in the binary expansion of N.

Example: Since $83 = 64 + 16 + 2 + 1$ has binary expansion (1010011), then row 83 has $\text{pow}(2, 4) = 16$ odd numbers.

Below is the implementation of above approach :

C++

```
// CPP code to find the count of odd numbers
// in n-th row of Pascal's Triangle
#include <bits/stdc++.h>
using namespace std ;

/* Function to get no of set
   bits in binary representation
   of positive integer n */
int countSetBits(int n)
{
    unsigned int count = 0;
    while (n)
    {
        count += n & 1;
        n >>= 1;
    }

    return count;
}

int countOfOddsPascal(int n)
{
    ---> // Count number of 1's in binary
         // representation of n.
    int c = countSetBits(n);

    // Number of odd numbers in n-th
    // row is 2 raised to power the count.
    return pow(2, c);
}

// Driver code
int main()
{
    int n = 20;
    cout << countOfOddsPascal(n) ;
    return 0;
}
```

Java

```
// Java code to find the count of odd
// numbers in n-th row of Pascal's
```

```
// Triangle
import java.io.*;

class GFG {

    /* Function to get no of set
    bits in binary representation
    of positive integer n */
    static int countSetBits(int n)
    {
        long count = 0;
        while (n > 0)
        {
            count += n & 1;
            n >>= 1;
        }

        return (int)count;
    }

    static int countOfOddsPascal(int n)
    {

        // Count number of 1's in binary
        // representation of n.
        int c = countSetBits(n);

        // Number of odd numbers in n-th
        // row is 2 raised to power the
        // count.
        return (int)Math.pow(2, c);
    }

    // Driver code
    public static void main (String[] args)
    {
        int n = 20;
        System.out.println(
            countOfOddsPascal(n));
    }
}

// This code is contributed by anuj_67.
```

Python3

```
# Python code to find the count of
# odd numbers in n-th row of
# Pascal's Triangle

# Function to get no of set
# bits in binary representation
# of positive integer n
def countSetBits(n):
    count = 0
    while n:
        count += n & 1
```

```

        n >>= 1

    return count

def countOfOddPascal(n):

    # Count number of 1's in binary
    # representation of n.
    c = countSetBits(n)

    # Number of odd numbers in n-th
    # row is 2 raised to power the count.
    return pow(2, c)

# Driver Program
n = 20
print(countOfOddPascal(n))

# This code is contributed by Shrikant13

```

C#

```

// C# code to find the count of odd numbers
// in n-th row of Pascal's Triangle
using System;

class GFG {

    /* Function to get no of set
    bits in binary representation
    of positive integer n */
    static int countSetBits(int n)
    {
        int count = 0;
        while (n > 0)
        {
            count += n & 1;
            n >>= 1;
        }

        return count;
    }

    static int countOfOddsPascal(int n)
    {
        // Count number of 1's in binary
        // representation of n.
        int c = countSetBits(n);

        // Number of odd numbers in n-th
        // row is 2 raised to power the
        // count.
        return (int)Math.Pow(2, c);
    }

    // Driver code
    public static void Main ()

```

```
{
    int n = 20;
    Console.WriteLine(
        countOfOddsPascal(n)) ;
}

// This code is contributed by anuj_67.
```

PHP

```
<?php
// PHP code to find the
// count of odd numbers
// in n-th row of Pascal's
// Triangle

/* Function to get no of set
   bits in binary representation
   of positive integer n */
function countSetBits($n)
{
    $count = 0;
    while ($n)
    {
        $count += $n & 1;
        $n >>= 1;
    }

    return $count;
}

function countOfOddsPascal($n)
{
    --> // Count number of 1's in binary
        // representation of n.
        $c = countSetBits($n);

        // Number of odd numbers in n-th
        // row is 2 raised to power the count.
        return pow(2, $c);
}

// Driver code
$n = 20;
echo countOfOddsPascal($n) ;

// This code is contributed by mits.
?>
```

Output:

4

Time Complexity : $O(L)$, where L is the length of binary representation of given N .

Reference : <https://www.math.hmc.edu/funfacts/ffiles/30001.4-5.shtml>

广告 X

Learn coding by building games

JavaScript for Kids and Teens

Coding is fun with the right platform! Build and share amazing games with your friends.

codeguppy.com

OPEN

Recommended Posts:

Pascal's Triangle

Sum of all elements up to Nth row in a Pascal triangle

Calculate nCr using Pascal's Triangle

Leibniz harmonic triangle

XNOR of two numbers

Maximum XOR using K numbers from 1 to n

Sum of numbers with exactly 2 bits set

Equal Sum and XOR of three Numbers

Find two numbers from their sum and XOR

Check whether bitwise OR of N numbers is Even or Odd

Count numbers whose XOR with N is equal to OR with N

Find k numbers which are powers of 2 and have sum N | Set 1

Check whether product of ' n ' numbers is even or odd

Check if two numbers are bit rotations of each other or not

Count numbers whose sum with x is equal to XOR with x



MEETPATEL2

Check out this Author's [contributed articles](#).

If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your

article appearing on the GeeksforGeeks main page and help other Geeks.

Please Improve this article if you find anything incorrect by clicking on the "Improve Article" button below.

Improved By : [shrikanth13](#), [Mithun Kumar](#), [vt_m](#)

Article Tags : [Bit Magic](#) [binomial coefficient](#)

Practice Tags : [Bit Magic](#)



Be the First to upvote.

3.5

☐ To-do ☐ Done

Based on 2 vote(s)

[Feedback/ Suggest Improvement](#)

[Add Notes](#)

[Improve Article](#)

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

--->

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

A computer science portal for geeks

5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY

[About Us](#)
[Careers](#)
[Privacy Policy](#)
[Contact Us](#)

PRACTICE

[Courses](#)
[Company-wise](#)
[Topic-wise](#)
[How to begin?](#)

LEARN

[Algorithms](#)
[Data Structures](#)
[Languages](#)
[CS Subjects](#)
[Video Tutorials](#)

CONTRIBUTE

[Write an Article](#)
[Write Interview Experience](#)
[Internships](#)
[Videos](#)

@geeksforgeeks, Some rights reserved

--->

