

You are given two positive numbers M and N. The task is to print **greatest common divisor** of M'th and N'th **Fibonacci Numbers**.

3.5

The first few Fibonacci Numbers are 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144,

Note that 0 is considered as 0'th Fibonacci Number.

Examples:

Input : M = 3, N = 6
Output : 2
Fib(3) = 2, Fib(6) = 8
GCD of above two numbers is 2

Input : M = 8, N = 12
Output : 3
Fib(8) = 21, Fib(12) = 144
GCD of above two numbers is 3

Recommended: Please solve it on "PRACTICE" first, before moving on to the solution.

A **Simple Solution** is to follow below steps.

- 1) Find M'th Fibonacci Number.
- 2) Find N'th Fibonacci Number.
- 3) Return GCD of two numbers.

A **Better Solution** is based on below identity

$$\text{GCD}(\text{Fib}(M), \text{Fib}(N)) = \text{Fib}(\text{GCD}(M, N))$$

The above property holds because Fibonacci Numbers follow Divisibility Sequence, i.e., if M divides N, then Fib(M) also divides N. For example, Fib(3) = 2 and every third third Fibonacci Number is even.

Source : [Wiki](#)

The steps are:

- 1) Find GCD of M and N. Let GCD be g.
- 2) Return Fib(g).

Below are implementations of above idea.

C++

```
// C++ Program to find GCD of Fib(M) and Fib(N)
#include <bits/stdc++.h>
using namespace std;
const int MAX = 1000;

// Create an array for memoization
int f[MAX] = {0};

// Returns n'th Fibonacci number using table f[].
// Refer method 6 of below post for details.
// https://www.geeksforgeeks.org/program-for-nth-fibonacci-number/
int fib(int n)
{
    // Base cases
    if (n == 0)
        return 0;
    if (n == 1 || n == 2)
        return (f[n] = 1);

    // If fib(n) is already computed
    if (f[n])
        return f[n];

    int k = (n & 1)? (n+1)/2 : n/2;

    // Applying recursive formula [Note value n&1 is 1
    // if n is odd, else 0.
    f[n] = (n & 1)? (fib(k)*fib(k) + fib(k-1)*fib(k-1))
        : (2*fib(k-1) + fib(k))*fib(k);

    return f[n];
}

// Function to return gcd of a and b
int gcd(int M, int N)
{
    if (M == 0)
        return N;
    return gcd(N%M, M);
}

// Returns GCD of Fib(M) and Fib(N)
int findGCDofFibMFibN(int M, int N)
{
    return fib(gcd(M, N));
}

// Driver code
int main()
{
    int M = 3, N = 12;
    cout << findGCDofFibMFibN(M, N);
    return 0;
}
```

[Run on IDE](#)

Java

```
// Java Program to find GCD of Fib(M) and Fib(N)
class gcdOfFibonacci
{
    static final int MAX = 1000;
    static int[] f;

    gcdOfFibonacci() // Constructor
    {
        // Create an array for memoization
        f = new int[MAX];
    }

    // Returns n'th Fibonacci number using table f[].
    // Refer method 6 of below post for details.
    // https://www.geeksforgeeks.org/program-for-nth-fibonacci-number/
    private static int fib(int n)
    {
        // Base cases
        if (n == 0)
            return 0;
        if (n == 1 || n == 2)
            return (f[n] = 1);

        // If fib(n) is already computed
        if (f[n] != 0)
            return f[n];

        int k = ((n & 1) == 1) ? (n+1)/2 : n/2;

        // Applying recursive formula [Note value n&1 is 1
        // if n is odd, else 0.
        f[n] = ((n & 1) == 1) ? (fib(k)*fib(k) + fib(k-1)*fib(k-1))
            : (2*fib(k-1) + fib(k))*fib(k);

        return f[n];
    }

    // Function to return gcd of a and b
    private static int gcd(int M, int N)
    {
        if (M == 0)
            return N;
        return gcd(N%M, M);
    }

    // This method returns GCD of Fib(M) and Fib(N)
    static int findGCDofFibMFibN(int M, int N)
    {
        return fib(gcd(M, N));
    }

    // Driver method
    public static void main(String[] args)
    {
        // Returns GCD of Fib(M) and Fib(N)
        gcdOfFibonacci obj = new gcdOfFibonacci();
        int M = 3, N = 12;
        System.out.println(findGCDofFibMFibN(M, N));
    }
}
// This code is contributed by Pankaj Kumar
```

[Run on IDE](#)

Output:

2

This article is contributed by **Shubham Agrawal**. 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 write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Watch
and

Watch
and

GATE CS Corner Company Wise Coding Practice

Mathematical Fibonacci GCD-LCM

[Login to Improve this Article](#)

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

Recommended Posts:

[Program for Fibonacci numbers](#)

[Cassini's Identity](#)

[GCD of more than two \(or array\) numbers](#)

[Euclidean algorithms \(Basic and Extended\)](#)

[Finding number of digits in n'th Fibonacci number](#)

[Evaluation of Risk in Investments](#)

[Divide 1 to n into two groups with minimum sum difference](#)

[Check whether given three numbers are adjacent primes](#)

[Possible to make a divisible by 3 number using all digits in an array](#)

[Max occurring divisor in an interval](#)

([Login](#) to Rate)

3.5 Average Difficulty : **3.5/5.0**
Based on **4** vote(s)

Basic

Easy

Medium

Hard

Expert



Add to TODO List



Mark as DONE

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

Load Comments

Share this post!

@geeksforgeeks, Some rights reserved

Policy



Contact Us!

About Us!

Careers!

Privacy