Home Algorithms DS GATE Interview Corner Q&A C C++ Java Books Contribute Ask a Q About
Array Bit Magic C/C++ Articles GFacts Linked List MCQ Misc Output String Tree Graph

# Program for Fibonacci numbers

The Fibonacci numbers are the numbers in the following integer sequence.

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 141, ……..

In mathematical terms, the sequence Fn of Fibonacci numbers is defined by the recurrence relation

$$F_n = F_{n-1} + F_{n-2}$$

with seed values

$$F_0 = 0 \quad \text{and} \quad F_1 = 1.$$

Write a function *int fib(int n)* that returns $F_n$. For example, if *n* = 0, then *fib()* should return 0. If n = 1, then it should return 1. For n > 1, it should return $F_{n-1} + F_{n-2}$

Following are different methods to get the nth Fibonacci number.

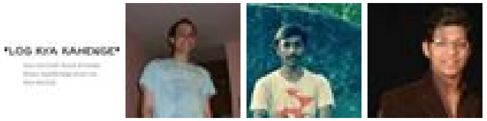
**Method 1 ( Use recursion )**
A simple method that is a direct recusrive implementation mathematical recurance relation given above.

```
#include<stdio.h>
int fib(int n)
{
   if (n <= 1)
      return n;
   return fib(n-1) + fib(n-2);
}
```

Google™ Custom Search

GeeksforGeeks

53,523 people like GeeksforGeeks.



Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion

```
int main ()
{
  int n = 9;
  printf("%d", fib(n));
  getchar();
  return 0;
}
```

*Time Complexity:* T(n) = T(n-1) + T(n-2) which is exponential.
We can observe that this implementation does a lot of repeated work (see the following recursion tree). So this is a bad implementation for nth Fibonacci number.

```
                         fib(5)
                     /             \
               fib(4)                fib(3)
             /      \                /     \
         fib(3)      fib(2)         fib(2)    fib(1)
        /     \        /    \       /    \
  fib(2)   fib(1)  fib(1) fib(0) fib(1) fib(0)
  /    \
fib(1) fib(0)
```

*Extra Space:* O(n) if we consider the fuinction call stack size, otherwise O(1).

**Method 2 ( Use Dynamic Programming )**
We can avoid the repeated work done is the method 1 by storing the Fibonacci numbers calculated so far.

```
#include<stdio.h>

int fib(int n)
{
  /* Declare an array to store fibonacci numbers. */
  int f[n+1];
  int i;

  /* 0th and 1st number of the series are 0 and 1*/
  f[0] = 0;
  f[1] = 1;

  for (i = 2; i <= n; i++)
  {
```

Geometric Algorithms

Popular Posts

All permutations of a given string
Memory Layout of C Programs
Understanding “extern” keyword in C
Median of two sorted arrays
Tree traversal without recursion and without stack!
Structure Member Alignment, Padding and Data Packing
Intersection point of two Linked Lists
Lowest Common Ancestor in a BST.
Check if a binary tree is BST or not
Sorted Linked List to Balanced BST

```
      /* Add the previous 2 numbers in the series
         and store it */
      f[i] = f[i-1] + f[i-2];
  }

  return f[n];
}

int main ()
{
  int n = 9;
  printf("%d", fib(n));
  getchar();
  return 0;
}
```

*Time Complexity:* O(n)

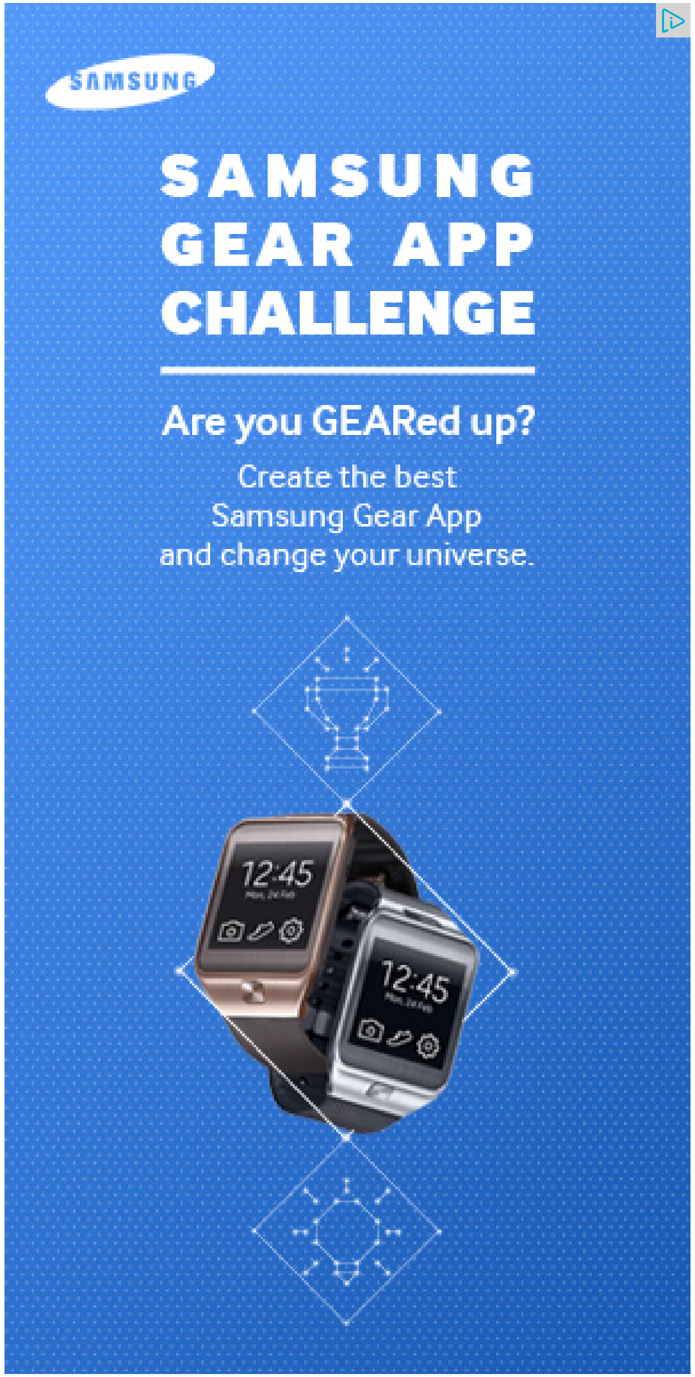
*Extra Space:* O(n)

**Method 3 ( Space Otimized Method 2 )**

We can optimize the space used in method 2 by storing the previous two numbers only because that is all we need to get the next Fibannaci number in series.

```
#include<stdio.h>
int fib(int n)
{
  int a = 0, b = 1, c, i;
  if( n == 0)
    return a;
  for (i = 2; i <= n; i++)
  {
     c = a + b;
     a = b;
     b = c;
  }
  return b;
}

int main ()
{
  int n = 9;
  printf("%d", fib(n));
  getchar();
  return 0;
}
```


SAMSUNG
SAMSUNG GEAR APP CHALLENGE
Are you GEARed up?
Create the best Samsung Gear App and change your universe.
12:45
12:45

*Time Complexity:* O(n)
*Extra Space:* O(1)

**Method 4 ( Using power of the matrix {{1,1},{1,0}} )**
This another O(n) which relies on the fact that if we n times multiply the matrix M = {{1,1},{1,0}} to itself (in other words calculate power(M, n )), then we get the (n+1)th Fibonacci number as the element at row and column (0, 0) in the resultant matrix.

The matrix representation gives the following closed expression for the Fibonacci numbers:

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n = \begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix}.$$

```
#include <stdio.h>

/* Helper function that multiplies 2 matricies F and M of size 2*2, an
  puts the multiplication result back to F[][] */
void multiply(int F[2][2], int M[2][2]);

/* Helper function that calculates F[][] raise to the power n and puts
  result in F[][]
  Note that this function is desinged only for fib() and won't work as
  power function */
void power(int F[2][2], int n);

int fib(int n)
{
  int F[2][2] = {{1,1},{1,0}};
  if (n == 0)
      return 0;
  power(F, n-1);

  return F[0][0];
}

void multiply(int F[2][2], int M[2][2])
{
  int x =  F[0][0]*M[0][0] + F[0][1]*M[1][0];
  int y =  F[0][0]*M[0][1] + F[0][1]*M[1][1];
  int z =  F[1][0]*M[0][0] + F[1][1]*M[1][0];
  int w =  F[1][0]*M[0][1] + F[1][1]*M[1][1];

  F[0][0] = x;
  F[0][1] = y;
  F[1][0] = z;
```

## Recent Comments

**Aman** Hi, Why arent we checking for conditions...
Write a C program to Delete a Tree. · 37 minutes ago

kzs please provide solution for the problem...
Backtracking | Set 2 (Rat in a Maze) · 41 minutes ago

**Sanjay Agarwal** bool tree::Root_to_leaf_path_given_sum(tree...
Root to leaf path sum equal to a given number · 1 hour ago

**GOPI GOPINATH** @admin Highlight this sentence "We can easily...
Count trailing zeroes in factorial of a number · 1 hour ago

**newCoder3006** If the array contains negative numbers also. We...
Find subarray with given sum · 1 hour ago

**newCoder3006** Code without using while loop. We can do it...
Find subarray with given sum · 1 hour ago

```
  F[1][1] = w;
}

void power(int F[2][2], int n)
{
  int i;
  int M[2][2] = {{1,1},{1,0}};

  // n - 1 times multiply the matrix to {{1,0},{0,1}}
  for (i = 2; i <= n; i++)
      multiply(F, M);
}

/* Driver program to test above function */
int main()
{
  int n = 9;
  printf("%d", fib(n));
  getchar();
  return 0;
}
```

*Time Complexity:* O(n)
*Extra Space:* O(1)

**Method 5 ( Optimized Method 4 )**
The method 4 can be optimized to work in O(Logn) time complexity. We can do recursive multiplication to get power(M, n) in the prevous method (Similar to the optimization done in this post)

```
#include <stdio.h>

void multiply(int F[2][2], int M[2][2]);

void power(int F[2][2], int n);

/* function that returns nth Fibonacci number */
int fib(int n)
{
  int F[2][2] = {{1,1},{1,0}};
  if (n == 0)
    return 0;
  power(F, n-1);
  return F[0][0];
}

/* Optimized version of power() in method 4 */
void power(int F[2][2], int n)
{
  if( n == 0 || n == 1)
      return;
  int M[2][2] = {{1,1},{1,0}};

  power(F, n/2);
  multiply(F, F);

  if (n%2 != 0)
     multiply(F, M);
}

void multiply(int F[2][2], int M[2][2])
{
  int x =  F[0][0]*M[0][0] + F[0][1]*M[1][0];
  int y =  F[0][0]*M[0][1] + F[0][1]*M[1][1];
  int z =  F[1][0]*M[0][0] + F[1][1]*M[1][0];
  int w =  F[1][0]*M[0][1] + F[1][1]*M[1][1];

  F[0][0] = x;
  F[0][1] = y;
  F[1][0] = z;
  F[1][1] = w;
}

/* Driver program to test above function */
int main()
{
  int n = 9;
  printf("%d", fib(9));
  getchar();
  return 0;
}
```

***Time Complexity:*** **O(Logn)**

*Extra Space:* O(Logn) if we consider the function call stack size, otherwise O(1).

Please write comments if you find the above codes/algorithms incorrect, or find other ways to solve the same problem.

**References:**

http://en.wikipedia.org/wiki/Fibonacci_number
http://www.ics.uci.edu/~eppstein/161/960109.html

## Related Tpoics:

- Backtracking | Set 8 (Solving Cryptarithmetic Puzzles)
- Tail Recursion
- Find if two rectangles overlap
- Analysis of Algorithm | Set 4 (Solving Recurrences)
- Print all possible paths from top left to bottom right of a mXn matrix
- Generate all unique partitions of an integer
- Russian Peasant Multiplication
- Closest Pair of Points | O(nlogn) Implementation

55 Tweet 5 4

**Writing code in comment?** Please use **ideone.com** and share the link here.

**57 Comments** **GeeksforGeeks**

Sort by Newest

Join the discussion…

**Wei Xue** • 2 days ago

The time complexity of dynamic programming is NOT O(n). It looks O(n), but a
larger, the operation of addition will increase, which is not O(1).

**Pegasi** • 4 months ago

The 12th fib number is 144 apparently

**samtron92** → Pegasi • 9 days ago

true,, 141 is incorrect it should be 144

**Guest** → Pegasi • 9 days ago

true, 141 is incorrect

**s.a.** • 4 months ago

can the last method be used to print the fibonacci series i.e. in (log n) time ? if

1

**Pegasi** → s.a. • 4 months ago

Yes the recursion does it in O(log n) time. Here is an iterative power fu

```
void power(int F[2][2], int n)
{
int P[2][2] = {{1,1},{1,0}};
if (n % 2 != 0)
multiply(P, F);
while (n > 0)
{
n /= 2;
multiply(F, F);
}
}
```

**s.a.** • 4 months ago

can the last method be used to print the fibonacci series i.e. in (nlog n) time ?

**J Reyes** • 6 months ago

In Java

```
public static long fib(int n) {
if(n == 0) { return n; }
n = n - 1;

BigDecimal eigen1 = new BigDecimal("-.61803398875");
BigDecimal eigen2 = new BigDecimal("1.61803398875");
BigDecimal det = new BigDecimal("-.4472135955" );
BigDecimal fib = eigen1.pow(n+1).subtract(eigen2.pow(n+1)).multiply(det);

return fib.longValue();
}
```

**A Friend from hiddle leaf** • 6 months ago

comment section sucks cant write while loop

**A Friend from hiddle leaf** · 6 months ago

Comment section sucks

```
concept-
int t0=1;
it t1=1;
cout<<t0<<endl; while(t1<20000)="" {="" cout<<t1<<endl;="" int="" temp="t1;"
```

**A Friend from hiddle leaf** · 6 months ago

//Take this losers, will run directly in Dev C++ without any change
//THIS-IS-NOOB

```
#include<iostream>
#include<conio.h>
using namespace std;
int main()
{
int maxRange;
cout<<"Hi there!">>endl;
cout<<"Enter the maximum range for fibonacci series:";
cin>>maxRange;
int to = 1;
int t1= 1;
cout<<t0<<endl; while(t1="" <="" maxrange)="" {="" cout<<t1<<endl;="" int=""
t0="temp;" }="" getch();="" return="" 0;="" }="">
```

**A Friend from hiddle leaf** · 6 months ago

```
#include<iostream>
#include<conio.h>
```

```
using namespace std;
int main()
{
int maxRange;
cout<<"Hi there!"<<endl; cout<<"enter="" the="" maximum="" range="" for=""
cin="">>maxRange;
int t0 = 1;
int t1 = 1;
cout<<t0<<endl; while(t1="" <="" maxrange)="" {="" cout<<t1<<endl;="" int=""
t0="temp;" }="" getch();="" return="" 0;="" }="" there="" was="" a="" problem="
formatting="">
```

**A Friend from hiddle leaf** • 6 months ago

```
//Take this losers, will run without any change in Dev C++
//THIS-IS-NOOB
#include<iostream>

#include<conio.h>

using namespace std;

int main()

{

int maxRange;

cout<<"Hi there!"<<endl; cout<<"enter="" the="" maximum="" range="" for=""
cin="">>maxRange;

int t0 = 1;

int t1 = 1;
```

cout<<t0<<endl; while(t1="" <="" maxrange)="" {="" cout<<t1<<endl;="" int=""
t0="temp;" }="" getch();="" return="" 0;="" }="">

**Karim** · 7 months ago

well there is another way, is to find where do we use fibonnaci, in the golden n
phi=(1+sqrt(5))/2
fib(n) = floor( phi^n/sqrt(5) +1/2)

2

**Sidhant** ➦ Karim · 3 months ago

Phi is irrational...so while coding u cant get it accurate enough..due to
numbers accurately

**aditya** · 8 months ago

plz sum1 post the solution of making program of fibonacci series using golder

1

**Ronny** · 10 months ago

@GeeksforGeeks

In method 4 statement and its description the fibonacci matrix is expressed as
whereas in the program follwing the description and the method 5 uses fibona

there is a typo
kindly update the post

**GeeksforGeeks** ➦ Ronny · 10 months ago

Thanks for pointing this out. We have corrected the typo.

**Ronny** ➦ GeeksforGeeks • 10 months ago

@GeeksforGeeks

There is still a typo in the description of the method 4.(only hea
needs to be corrected)

**GeeksforGeeks** ➦ Ronny • 10 months ago

Thanks Ronny, we have corrected it now.

**Kalyani Arla** • 10 months ago

if the callee function is above the caller function, you need not declare it(callee

**Mohammad Faizan Ali** • 11 months ago

5th solution is awesome.
keep up the very good work.

**Atiq Butt** • 11 months ago

0 0
1 1
2 10
3 101
4 10110
5 10110101
so on

2

**Atiq Butt** • 11 months ago

I need fibinoci of bit string like 0 for 0 1 for 1 but for 2 it must be 10 and for 3 it r

**Hardik Hadvani** • 11 months ago

Hey Adminr,

Excellent article for the Fibonacci series of course this blog is doing a very goo
I&#039m proud to be a part of its Readers community.
For the Fibonacci programs in different languange like C language, JAVA, C# r
http://www.hhhprogram.com/2013....

**Priyanka** • a year ago

What's the use of extra matrix M here when it's same as F. We can use F onl

```
/* Paste your code here (You may delete these lines if not writing c
```

**bohemia** ➦ Priyanka • a year ago

```
void multiply(int F[2][2], int M[2][2])
{
int x = F[0][0]*M[0][0] + F[0][1]*M[1][0];
int y = F[0][0]*M[0][1] + F[0][1]*M[1][1];
int z = F[1][0]*M[0][0] + F[1][1]*M[1][0];
int w = F[1][0]*M[0][1] + F[1][1]*M[1][1];

F[0][0] = x;
F[0][1] = y;
F[1][0] = z;
F[1][1] = w;
}
```

Maybe because in the above function , The Matrix F might have been n
power(F,n/2) , and obviously we need F*{{1,1},{1,0}} IF n doesn't happe
seperate {{1,1},{1,0}} as M ..Isn't it?

**Shivali Shakya** · a year ago

why you don&#039t declare a function?

**Manu Thakku** · a year ago

very helpful

**Rio Eduardo** · a year ago

How about this one? http://www.fansonnote.com/2012... Hope it will help

**pratheba** · a year ago

```
// source code c++

double fib1(int n)
{
double Phi = 1.618f;

double f = (std::pow(double(Phi),double(n)) - ( std::pow(double(-Phi),double(-n

int d = std::fmod(f,(double)1)* 10;
if( d >=5)
return std::ceil(f);
else
return std::floor(f);
}
```

int main()
{
int f1 = fib1(8);
std::cout << f1 << std::endl;
}

**pratheba** • a year ago

Fib(n) = (Phi^n - (-(Phi))^(-n))/(sqrt(5))
Phi = 1.618 ... (golden ratio )

http://www.maths.surrey.ac.uk/...

**Sameer023** • a year ago

Program making use of the below observation runs in O(log n) time (Concept

f(2n) = f(n)*f(n)+f(n+1)*f(n+1)
f(2n+1) = 2f(n)*f(n+1)+ f(n+1)*f(n+1)

Below C code is tested successfully.
Notation: f(1) = 0; f(2) = 1; f(3) = 1; .... and so on

```
#include<stdio.h>
#include<math.h>

main() {


int n, bit_seq, set_bit, f1, f2, f3,f4, count=0;

printf("Assuming the fibbonoci numbers start at index 1 \n");
printf("Enter a number: ");
```

see more

1

**Nishant** • 2 years ago

there is a typo at the second line
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 141..

the last number should be 144.

2

**atul** • 2 years ago

please correct the description.
you hav used m[][]={{1,0},{0,1}); in the explantion.
but code is using m[][]={{1,1},{1,0}}

```
/* Paste your code here (You may delete these lines if not writing co
```

**GeeksforGeeks** atul • 2 years ago

@atul: Thanks for pointing this out. We have corrected the matrix in ex

**iceman** • 2 years ago

you didn't need temp

```
 #include <stdio.h>
int power(int n,int m){
        if(m==0)
                return 1;
        else if(m&1){
                return power(n*n,(m-1)/2)*n;
        }
        else
                return power(n*n, m/2);
}
int main(int argc, char *argv[]){
        int n,m;
        while (scanf("%d%d", &n,&m)!=EOF) {
                printf("%d\n", power(n, m));
        }
        return 0;
}
```

**GeeksforGeeks** • 2 years ago

@Bhaskar: Thanks for suggesting a new method. We will test this method an

**Bhaskar** • 2 years ago

O(logn) code for computing fib(n)

```
#include <stdio.h>


long int fib(long int n) {

        long int a=1, b=0, p=0, q=1, prev_a, prev_p = 0;
        while(n>0) {
                if (n%2 == 0) {
                        prev_p = p;
                        p = p*p + q*q;
                        q = 2*prev_p*q + q*q;
                        n /= 2;
                } else {
                        prev_a = a;
```

see more

**Bhaskar** • 2 years ago

Another O(logn) :

```


#include <stdio.h>

long int fib(long int n) {

        long int a=1, b=0, p=0, q=1, prev_a, prev_p = 0;
        while(n>0) {
                if (n%2 == 0) {
                        prev_p = p;
                        p = p*p + q*q;
                        q = 2*prev_p*q + q*q;
                        n /= 2;
                } else {
                        prev_a = a;
                        a = b*q + a*q + a*p;
```

see more

**jia** • 3 years ago

In Method 3 :

one variable can be reduced in following way......

```
int fib(int n)
{
int a = 0, b = 1, i;
if( n == 0)
return a;
for (i = 2; i <= n; i++)
{
b=a+b;
a=b-a;

}
return b;
}

int main ()
{
int n = 4;
printf("%d", fib(n));
getchar();
return 0;
}
```

**mohan** • 3 years ago

f(2n) = f(n-1)*f(n)+f(n)*f(n+1)

f(2n+1) = f(n)*f(n)+f(n+1)*f(n+1)

so we can do this by log n with out any matrix multiplication

1 •

**Algoseekar** • 3 years ago

@geksforgeeks, venki can you prove mathematically that 5th method is Log(N

**Algoseekar** Algoseekar • 3 years ago

@sandeep,@ vanki,geeksforgeek..guys can you explain how complexi

**Sandeep** Algoseekar • 3 years ago

@Algoseeker: Following is the recurrance relation for method 5

T(n) = T(n/2) + O(1)

O(1) is there in the above expression because matrix multiplica
time.

This is a standard Binary Search Recurrance and solution of th

**Algoseekar** Sandeep • 3 years ago

@sandeep can u explain here nth means if n=0 is then
number or zeroth Fibonacci number..??

**Sandeep** Algoseekar • 3 years ago

It's the 0th Fibonacci number.

**Algoseekar** Sandeep • 3 years ago

@sandeep can u explain what actual optimization we a
optimization at which point..????

**wgpshashank** ➜ Algoseekar • 3 years ago

@AlgoSeekar , Dear Algoseekar Please Have Close Lo
Why we are doing the optimization ..this approach will s
approach for such question

We all know the Fibonacci recurrence as F(n+1) = F(n)
this in the form a matrix as shown below:

Look at the matrix A = [ [ 1 1 ] [ 1 0 ] ] . Multiplying A with
F(n+1) F(n) ] , so we say that

A* [ F(n) F(n-1) ] = [ F(n+1) F(n) ]

start with [ F(1) F(0) ] , multiplying it with A gives us [ F(
F(1) ] with A gives us [ F(3) F(2) ] and so on...

A* [ F(1) F(0) ] = [ F(2) F(1) ]
A* [ F(2) F(1) ] = [ F(3) F(2) ] = A^2 * [ F(1) F(0) ]
A* [ F(1) F(0) ] = [ F(4) F(3) ] = A^3 * [ F(1) F(0) ]
..

see more

**Algoseekar** ➜ Algoseekar • 3 years ago

i mean time complexity is O(logn)..how its comes

Load more comments

✉ Subscribe Ⓓ Add Disqus to your site