Name:Saurabh Mukherjee Roll no:001910501006 Class:BCSE II Sem :First Session 2020-21

Assignment Set :1

Problem No. 2

Problem Statement :

Write a program to generate the *nth* Fibonacci number iteratively and recursively. Check when there is overflow in the result and change the data types for accommodating higher values of inputs. Plot the Fibonacci number vs *n* graph.

Solution Approach:

Iterative Approach:

For calculating the nth Fibonacci number, firstly we have the 0th and 1st Fibonacci number which are 0 and 1 respectively which we store inside the variables a and b respectively and take a new variable to be c.We then run a for loop from 2 to n and each time update the value of c as the sum of a and b,a as b,b as c and return the value of c to get the nth Fibonacci number.

Recursive Approach:

The recursive approach has base case when n=0 ,Fibonacci(n) returns 0 and 1 when n=1.For the rest non-negative values of n we calculate Fibonacci(n) to be the sum of Fibonacci(n-1) and Fibonacci(n-2).We return the value.

Overflow Check:

In both the above cases the overflow is checked by continuously incrementing the values of n and finding the first mismatch of the actual result with that obtained from our program.

Structured Pseudocode :

Iterative Approach

Fibonacci(n)

1.Initialise a=0 ,b=1, c

2.For i from 2 to n:

3. c=a+b ,a=b, b=c

4. return c

Recursive Approach:

Fibonacci(n)

1.if n<=1

2. return n

3. else

4. return Fibonacci(n-1)+Fibonacci(n-2)

Results:

The results obtained from both of the above approaches depends on the data-type we are selecting for the return value from the Fibonacci function .For int the program is failing to give correct results at 47 and after that and for long or long long data type it is failing at 93 and deviating from the expected result.

Discussions:

The iterative approach is in this case is better and efficient than the recursive approach because the recursive approach grows exponentially with the input value n and a large number of overlapping subproblems appear which are solved again and again .In the iterative approach it takes linear time with the input value n and calculates the values much faster than the former approach .But ,in terms of accuracy after a certain value of n ,changing the data-type too would not work and we get wrong results due to overflow.

Separate files containing commented source code

The file has been attached.