ASSIGNMENT 6

Program 1: Fibonacci Series without using Recursion

Algorithm:

1. Input one=0, two=1, sum=0, i, n. Accept n from the user using scanf().

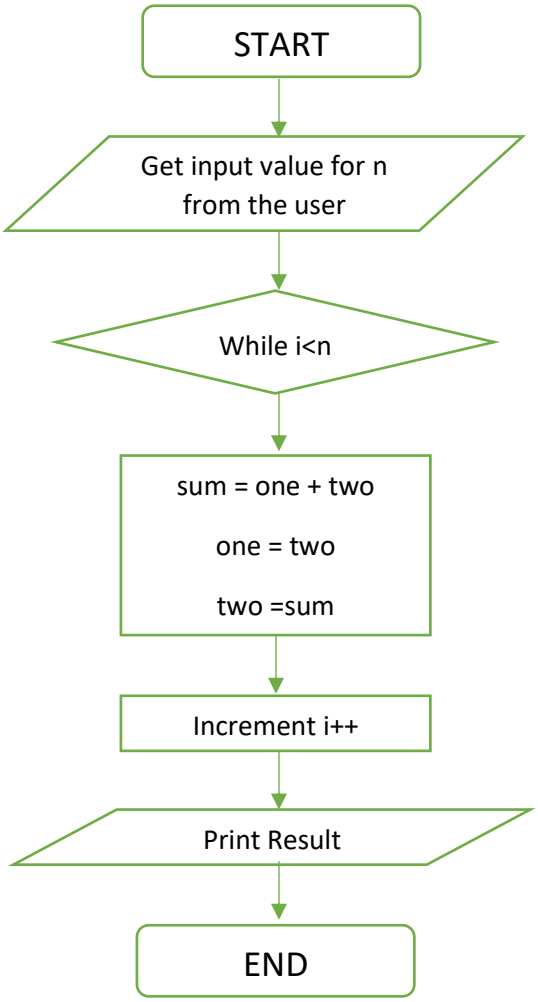
2. Print one and two.

3. Use a for loop where i=2 and let the loop iterate. Increment i in every iteration.

4. Next number is sum=one+two.

5. Print the series thus.

Flowchart:



Code:

```
#include <stdio.h>

int main() {
int i, n, one=0, two=1, sum=0;
printf("Enter the number of terms you want in a Fibonacci
sequence : ");
scanf("%d",&n);

printf("%d, %d, ", one, two);
i=2;
while(i<n)
{
    sum=one+two;
    one=two;
    two=sum;
    ++i;
printf("%d, ",sum);
}

    return 0;
}
```

Output:

```
Enter the number of terms you want in a Fibonacci sequence : 20

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597,
2584, 4181
```

Program 2: Fibonacci Series using Recursion

Algorithm:

1: Accept an input from user num = number of terms to be printed.

2. Declare function fibonacci ()

3.. Write a for loop with variable i, which should execute while i=0, i<=num, i++.

4. If n==0, then return 0.

5. If n==1, then return 1.

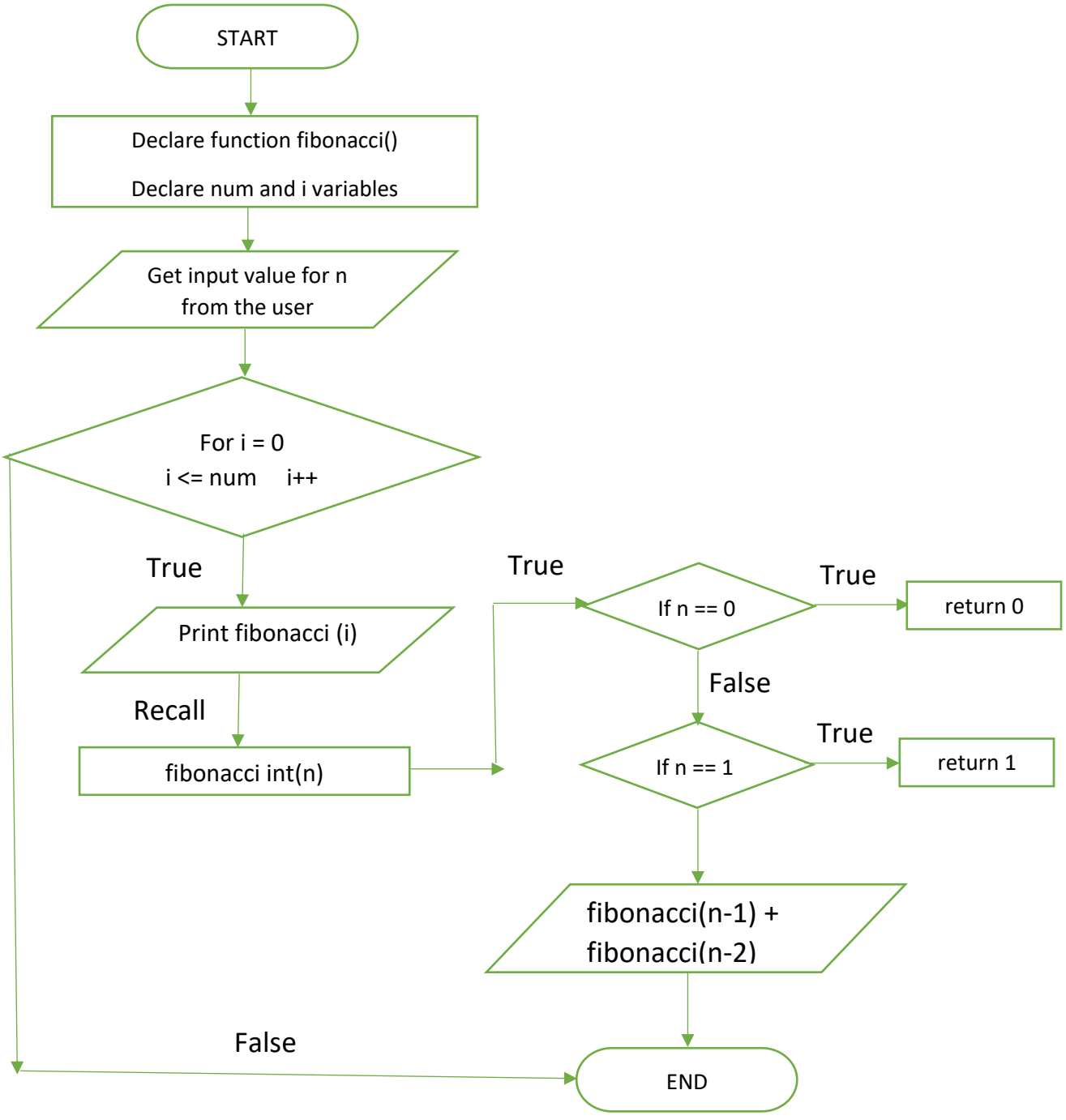
6. In function int fibonacci generate if loop for i<2 and return

the respective values.

7. Execution continues till loop condition is false.

8. Exit the program.

Flowchart:


START
Declare function fibonacci()
Declare num and i variables
Get input value for n from the user
For i = 0
i <= num i++
True
Print fibonacci (i)
Recall
fibonacci int(n)
True
If n == 0
True
return 0
False
If n == 1
True
return 1
fibonacci(n-1) + fibonacci(n-2)
False
END

Code:

```
#include <stdio.h>
int fibonacci(int num)
{
   if (num == 0)
   {
      return 0;
   }
   else if (num == 1)
   {
      return 1;
   }
   else
   {
      return fibonacci(num - 1) + fibonacci(num - 2);
   }
}


int main() {
   int num;
   printf("Enter the number of elements to be in the series: ");
   scanf("%d", &num);
```

```
    for (int i = 0; i < num; i++)
    {
        printf("%d, ", fibonacci(i));
    }
    return 0;
}
```

Output:

Enter the number of elements to be in the series: 8

0, 1, 1, 2, 3, 5, 8, 13,