**Name**: Ravi Jamanbhai Makwana

**Roll No**: 21BCP418

Div-6, G-12

**Assignment-4**

**1. Fibonacci Series**:

Write a program in any programming language to generate a Fibonacci series of ‘n’ numbers. Implement both the iterative method and recursive method.

Perform analysis of both the algorithms. For recursive method, generate the recurrence equation and solve it using substitution/induction method.

**Program:**

#include <stdio.h>

void Ifib(int n){

int a=0;

int b=1;

int counter=0;

while (counter<n){

printf("%d ",a);

int temp=a;

a=b;

b=temp+b;

counter++;

}

}

void Rfib(int n){

static int term1=0;

static int term2=1;

static int term3;

if (n>0){

term3 = term1 + term2;

term1 = term2;

term2 = term3;

printf("%d ",term3);

Rfib(n-1);

}

}

int main(){

int n;

printf("Enter the value of n: ");

scanf("%d",&n);

//Recursive Fibonacci function

printf("\nUsing Recursive method: ");

printf("%d %d ",0,1);

Rfib(n-2); //We have used n-2 because we have 2 numbers already printed here

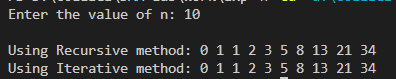
//Iterative Fibonacci function

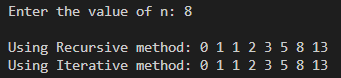
printf("\nUsing Iterative method: ");

Ifib(n);

}

**Output:**





**Analysis:**

**2. Binary Counter**:

Write a program in any programming language to increment the counter ‘n’ times, where n will be entered by user. Take suitable number of bits in counter depending upon the value of n, so that counter is incremented n times.

Perform worst case and amortized analysis of this algorithm.

**Program:**

#include <stdio.h>

#include <string.h>

void incrementer(char ch[]){

for (int i=strlen(ch)-1; i>=0; i--){

if (ch[i] == '0'){

ch[i] = '1';

return;

}

else if(ch[i] == '1'){

ch[i] = '0';

}

}

}

int main(){

int n;

printf("Enter n (number of increments) : ");

scanf("%d", &n);

char counter[n+1];

for (int i=0;i<n;i++){

counter[i] = '0';

}

counter[n] = '\0';

//incrementing the counter n times

for (int i=0; i<n; i++){

incrementer(counter);

}

printf("%s",counter);

}

**Output:**









**Analysis:**