DESIGN AND ANALYSIS OF ALGORITHMS

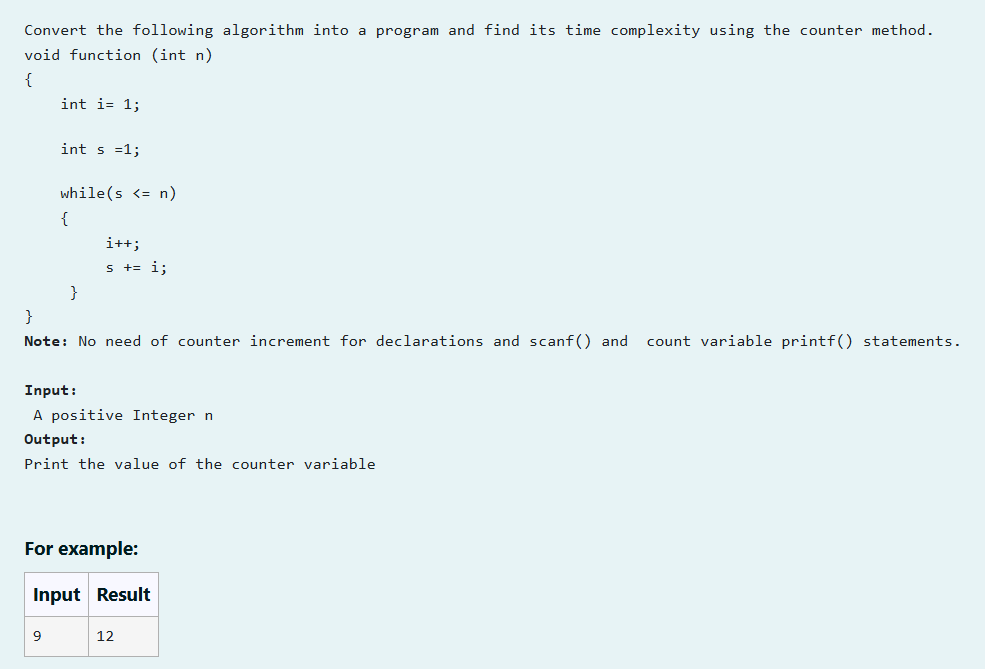
* ARITRA GUPTA

230701033

CSE-A

TIME COMPLEXITY

Ex 1.1

AIM:

SOURCE CODE:

#include<stdio.h>

void function(int n){

int count=0;

int i=1;

count++;

int s=1;

count++;

while(s<=n){

count++;

i++;

count++;

s=s+i;

count++;

}

count++;

printf("%d",count);

}

int main(){

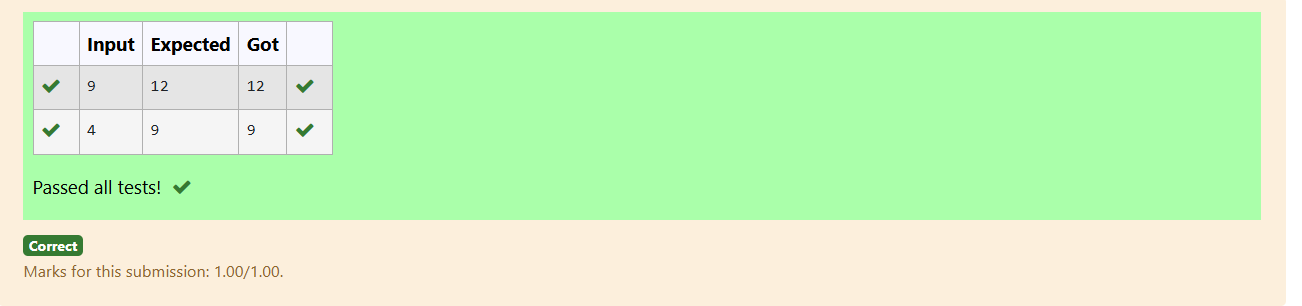
int n;

scanf("%d",&n);

function(n);

}

OUTPUT:

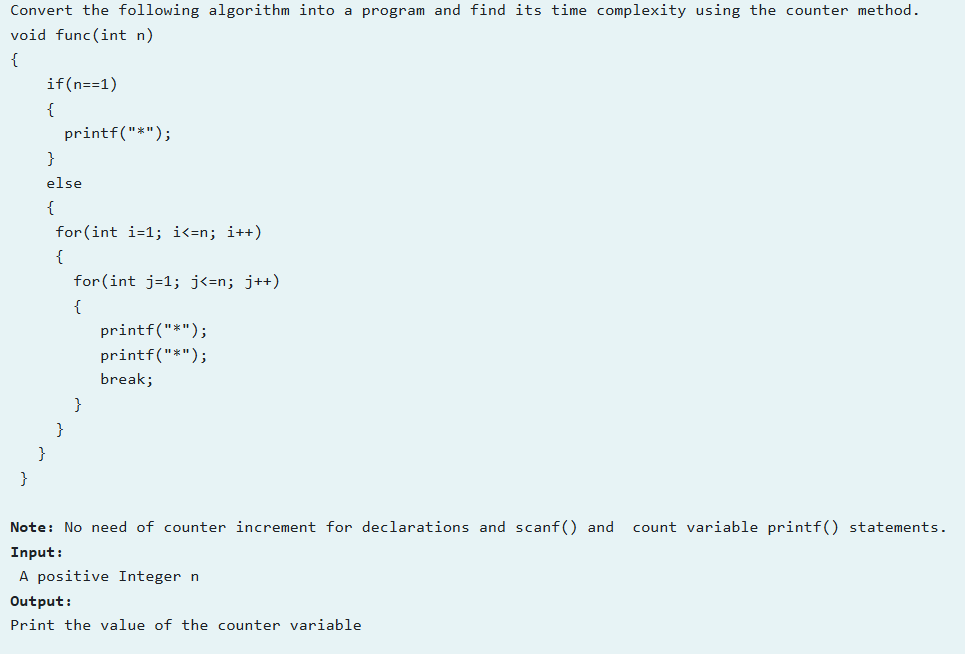
J

RESULT:

Thus the program was run successfully and output was verified.

EX 1.2

AIM:



SOURCE CODE:

#include<stdio.h>

void func(int n){

int count=0;

count++;

if(n==1){

}

else{

for(int i=1;i<=n;i++){

count=count+2;

count=count+2;

for(int j=1;j<=n;j++)

{

count++;

break;

}

}

count++;

}

printf("%d",count);

}

int main(){

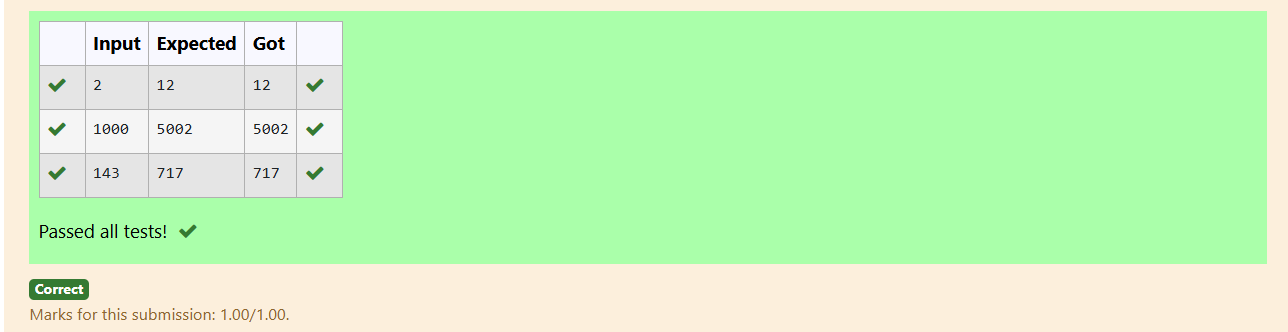
int n;

scanf("%d",&n);

func(n);

}

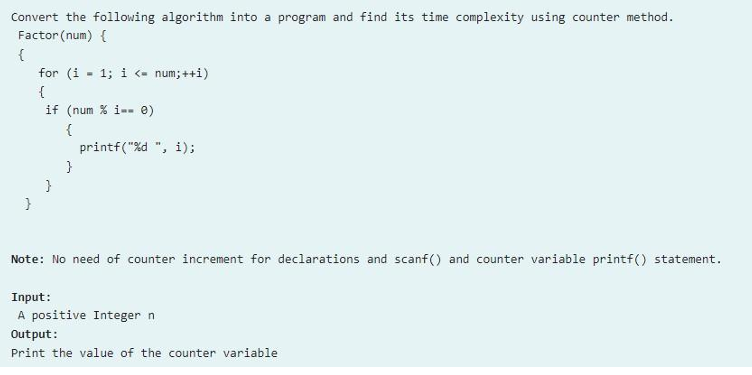
OUTPUT:



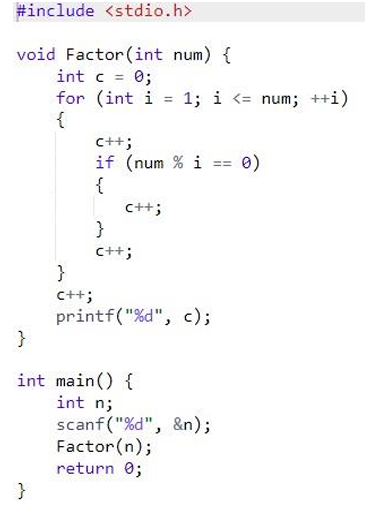
RESULT: Thus the above program was executed successfully and the output was verified.

EX 1.3

AIM:



SOURCE CODE:



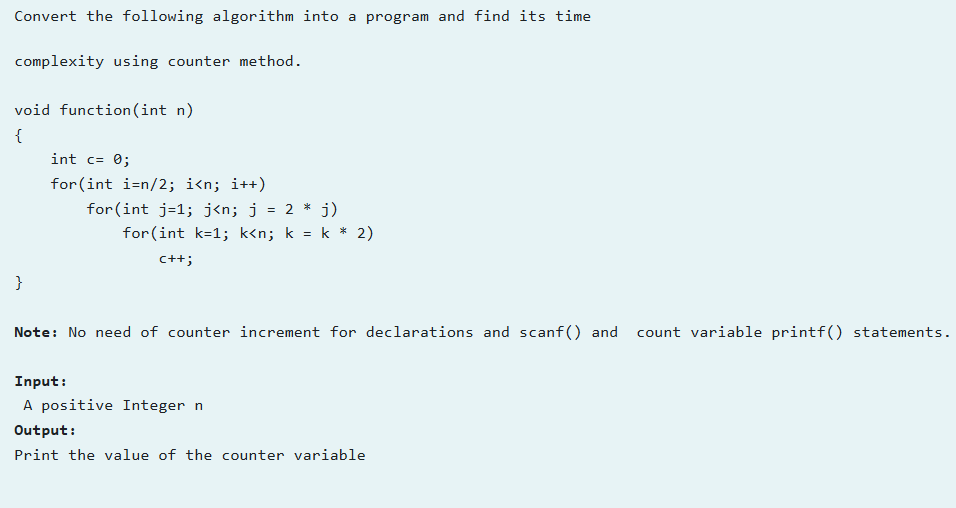
OUTPUT:



RESULT: Thus the program was run successfully and the output was verified .

Ex 2.4

AIM:



SOURCE CODE:

#include<stdio.h>

void function(int n)

{

int c= 0,count=0;

count++;//c=0

for(int i=n/2; i<n; i++)

{

count++;//fr loop1

for(int j=1; j<n; j = 2 \* j)

{

count++;//fr loop2

for(int k=1; k<n; k = k \* 2)

{

c++;

count+=2;//fr loop3 and c++

}

count++;//false1

}

count++;//false2

}

count++;//false3

printf("%d",count);

}

int main()

{

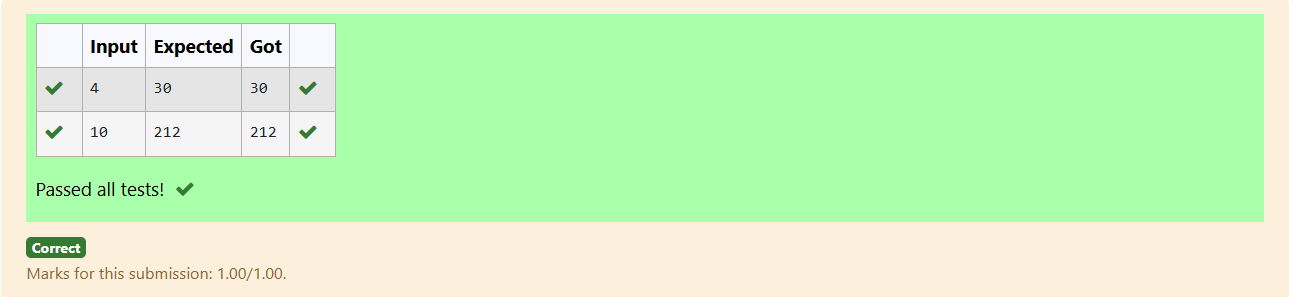
int n;

scanf("%d",&n);

function(n);

}

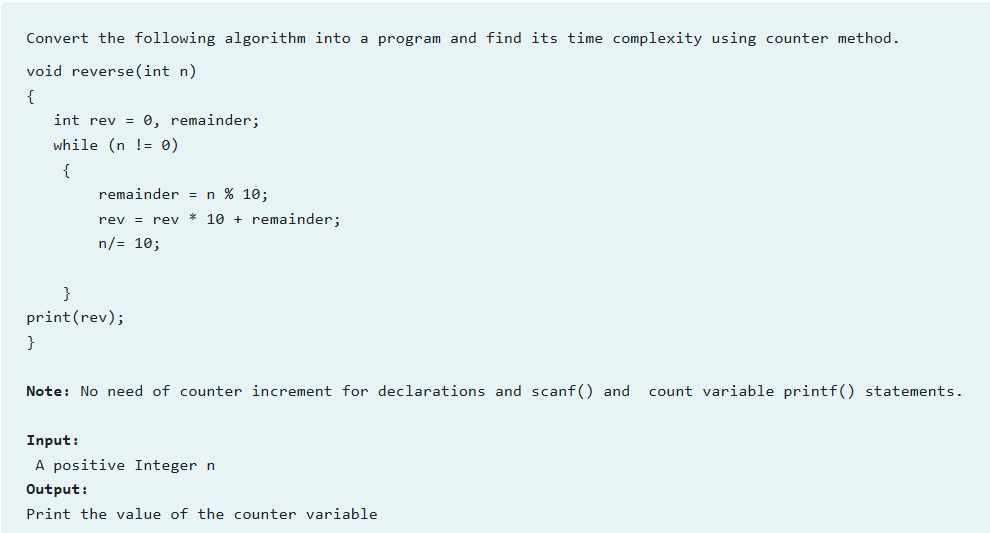
OUTPUT:



RESULT: Thus the program was executed successfully and the output was verified.

EX 2.5

AIM:



SOURCE CODE:

#include<stdio.h>

void reverse(int n)

{

int rev = 0, remainder,count=0;

count++;

while (n != 0)

{

count++;

remainder = n % 10;

rev = rev \* 10 + remainder;

n/= 10;

count+=3;

}

count++;

//print(rev);

count++;

printf("%d",count);

}

int main()

{

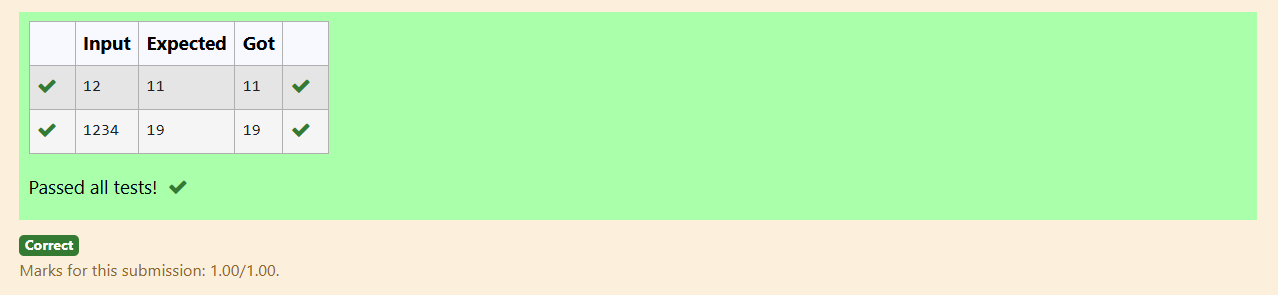
int n;

scanf("%d",&n);

reverse(n);

}

OUTPUT:



RESULT: Thus the program was executed successfully and the output was verified.