Week 1

Program - 1

**Aim:**

To convert the given algorithm into a C program and determine its time complexity using the counter method.

**Input:**

A positive integer n.

**Coding:**

#include <stdio.h>

void function(int n) {

int count = 0;

int i = 1;

count++;

int s = 1;

count++;

while (s <= n) {

i++; count++;

s += i; count++;

count++;

}

count++;

printf("%d", count);

}

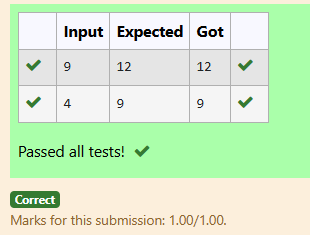
int main() {

int n;

scanf("%d", &n);

function(n);

}



Program – 2

**Aim:**

To convert the given algorithm into a C program and determine its time complexity using the counter method.

**Input:**

A positive integer n.

**Code:**

#include<stdio.h>

void func(int n)

{

int count=0;

count++;

if(n==1)

{

count=count;

}

else

{

count++;

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

{

count++;

count++;

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

{

count++;

count++;

break;

}

count++;

}

count++;

}

printf("%d",count-1);

}

int main(){

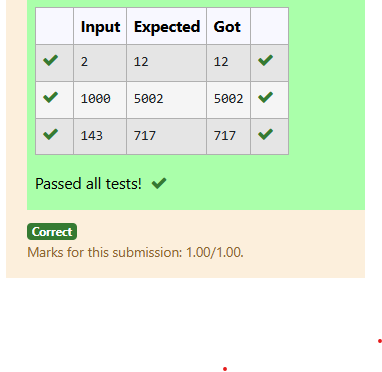
int n;

scanf("%d",&n);

func(n);

}

Output:



**Program – 3**

**Aim:**

To implement a program to find the factors of a given positive integer nnn and determine its time complexity using a counter method to count the number of iterations performed.

**Input:**

A positive integer nnn.

**Code:**

#include<stdio.h>

void Factor(int num)

{

int count=0;

int i;

for (i = 1; i <= num;++i)

{

count++;

count++;

if (num % i== 0)

{

count++;

//printf("%d ", i);

}

}

count++;

printf("%d",count);

}

int main(){

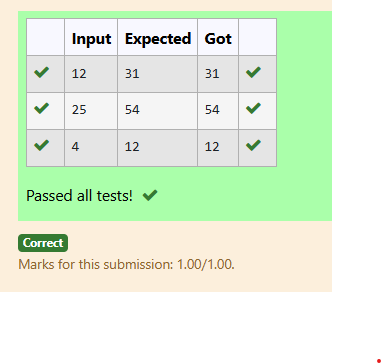
int num;

scanf("%d",&num);

Factor(num);

}

**Output:**

****

**Program 4:**

**Aim:**

To implement a program to calculate the number of iterations performed by the nested loops in the given algorithm using a counter method and determine its time complexity.

**Input:**

A positive integer nnn.

**Program:**

#include<stdio.h>

void function(int n)

{

int count=0;

count++;

int c= 0;

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

count++;

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

count++;

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

count++;

c++;

count++;

}

count++;

}

count++;

}

count++;

printf("%d",count);

}

int main(){

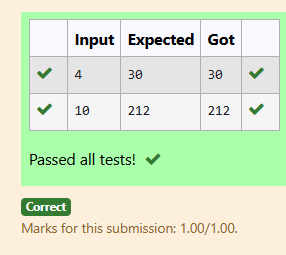
int n;

scanf("%d",&n);

function(n);

}

**Output:**

****

**Program – 5:**

**Aim:**

To implement a program to reverse the digits of a given positive integer nnn and determine the time complexity using a counter method.

**Input:**

A positive integer n.

Code:

#include<stdio.h>

void reverse(int n)

{

int count=0;

int rev = 0, remainder;

count++;

while (n != 0)

{

count++;

remainder = n % 10;

count++;

rev = rev \* 10 + remainder;

count++;

n/= 10;

count++;

}

count++;

//printf("%d",rev);

count++;

printf("%d",count);

}

int main(){

int n;

scanf("%d",&n);

reverse(n);

}

Output:

