Started on	Tuesday, 20 August 2024, 1:31 PM
State	Finished
Completed on	Tuesday, 20 August 2024, 1:45 PM
Time taken	14 mins 31 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

```
Convert the following algorithm into a program and find its time complexity using the counter method.
void function (int n)
{
    int i= 1;
```

```
    int s =1;
```

```
    while(s <= n)
    {
         i++;
         s += i;
     }
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:
A positive Integer n
Output:
Print the value of the counter variable

For example:

Input	Result
9	12

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
int n;
int count=0;
scanf("%d",&n);
int i=1;
count++;
int s=1;
count++;
while(s<=n)
{
count++;
i++;
count++;
s=s+i;
count++;
}
count++;
printf("%d",count);
return 0;
}
```

	Input	Expected	Got	
✔	9	12	12	✔
✔	4	9	9	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ BASIC C PROGRAMMING-PRACTICE

Jump to...

Problem 2: Finding Complexity using Counter method ►

Started on	Tuesday, 20 August 2024, 1:47 PM
State	Finished
Completed on	Tuesday, 20 August 2024, 2:06 PM
Time taken	18 mins 41 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using the counter method.

```
void func(int n)
{
    if(n==1)
    {
      printf("*");
    }
    else
    {
     for(int i=1; i<=n; i++)
     {
       for(int j=1; j<=n; j++)
       {
          printf("*");
          printf("*");
          break;
       }
     }
   }
 }
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

Print the value of the counter variable

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
    int n;
    int count=0;
    scanf("%d",&n);
    if(n==1){
        count++;
        //printf("");
        count++;
    }
    else
    {
     count++;
     for(int i=1; i<=n; i++)
     {
       count++;
       for(int j=1; j<=n; j++)
       {
          count++;
          //printf("");
          count++;
          //printf("");
          count++;
          break;
       }
       count++;
     }
     count++;
   }
   printf("%d",count);
 }
```

```
33
34
```

	Input	Expected	Got	
✔	2	12	12	✔
✔	1000	5002	5002	✔
✔	143	717	717	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ Problem 1: Finding Complexity using Counter Method

Jump to...

Problem 3: Finding Complexity using Counter Method ►

Started on	Tuesday, 20 August 2024, 2:07 PM
State	Finished
Completed on	Tuesday, 20 August 2024, 2:44 PM
Time taken	37 mins 36 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using counter method.

```
Factor(num) {
{
   for (i = 1; i <= num;++i)
   {
    if (num % i== 0)
       {
         printf("%d ", i);
       }
    }
 }
```

Note: No need of counter increment for declarations and scanf() and counter variable printf() statement.

Input:
A positive Integer n
Output:
Print the value of the counter variable

Answer:

```c
#include <stdio.h>

int main() {
    int num;
    int counter = 0;
    scanf("%d", &num);
    for (int i = 1; i <= num; ++i) {
        counter++;
        counter++;
        if (num % i == 0) {
            //printf("%d ", i);
            counter++;
        }
    }
    counter++;

    printf("%d", counter);

    return 0;
}
```

	Input	Expected	Got	
✔	12	31	31	✔
✔	25	54	54	✔
✔	4	12	12	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ Problem 2: Finding Complexity using Counter method

Jump to...

Problem 4: Finding Complexity using Counter Method ►

Started on	Tuesday, 20 August 2024, 2:45 PM
State	Finished
Completed on	Tuesday, 20 August 2024, 2:51 PM
Time taken	5 mins 54 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using counter method.

```
void function(int n)
{
    int c= 0;
    for(int i=n/2; i<n; i++)
        for(int j=1; j<n; j = 2 * j)
            for(int k=1; k<n; k = k * 2)
                c++;
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:
A positive Integer n

Output:
Print the value of the counter variable

Answer:

```c
#include<stdio.h>
int main()
{
    int c=0,n;
    int count=0;
    count++;
    scanf("%d",&n);
    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);
}
```

	Input	Expected	Got	
✔	4	30	30	✔
✔	10	212	212	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ Problem 3: Finding Complexity using Counter Method

Jump to...

Problem 5: Finding Complexity using counter method ►

Started on	Tuesday, 20 August 2024, 2:51 PM
State	Finished
Completed on	Monday, 26 August 2024, 7:26 PM
Time taken	6 days 4 hours
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using counter method.

```
void reverse(int n)
{
   int rev = 0, remainder;
   while (n != 0)
    {
        remainder = n % 10;
        rev = rev * 10 + remainder;
        n/= 10;

    }
print(rev);
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:
A positive Integer n
Output:
Print the value of the counter variable

Answer:

```c
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    int c =0;
    int rev =0,remainder;
    c++;
    while(n!=0)
    {c++;
     remainder = n % 10;
     c++;
     rev = rev * 10 + remainder;
     c++;
      n/= 10;
      c++;
}
c++;
//print(rev);
c++;
printf("%d",c);
}
```

	Input	Expected	Got	
✔	12	11	11	✔
✔	1234	19	19	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ Problem 4: Finding Complexity using Counter Method

Jump to...

1-Number of Zeros in a Given Array ►