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>
void function(int n);
int main(){
    int n;
    scanf("%d",&n);
    function(n);
}
void function(int n){
    int count=0;
    int i=1;
    count++;
    int s=1;
    count++;
    while(s<=n){
        count++;
        i++;
        count++;
        s+=i;
        count++;
    }
    count++;
    printf("%d",count);
}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

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>
void func(int n)
{
    int c=0;
    c++;
    if(n==1)
    {
        c++;
        //printf("*");

    }

    else
    {
     for(int i=1; i<=n; i++)
     {
         c++;
         for(int j=1; j<=n; j++)
         {
             c++;
             //printf("*");
             c++;
             //printf("*");
             c++;
             break;

         }
         c++;
     }
     c++;
   }
   printf("%d",c);
}
int main()
{
    int n;
    scanf("%d",&n);
    func(n);
}
```

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:

```
#include<stdio.h>
void Factor(int num)
{
    int c=0;

    for (int i = 1; i <= num;++i)
    {
        c++;
        if (num % i== 0)
        {

            //printf("%d ", i);
            c++;
        }
        c++;

    }
    c++;
    printf("%d",c);

}
int main()
{
    int num;
    scanf("%d",&num);
    Factor(num);
}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

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:

```
#include<stdio.h>
void function(int n)
{
    int ct=0;
    int c= 0;
    ct++;
    for(int i=n/2; i<n; i++)
    {
        ct++;
        for(int j=1; j<n; j = 2 * j)
        {
            ct++;
            for(int k=1; k<n; k = k * 2)
            {
                ct++;
                c++;
                ct++;
            }
            ct++;
        }
        ct++;

    }
    ct++;
    printf("%d",ct);

}
int main()
{
    int n;
    scanf("%d",&n);
    function(n);
}
```

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

Passed all tests! ✔

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:

```
#include<stdio.h>
void reverse(int 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++;
    c++;
    printf("%d",c);

//printf(rev);
}
int main()
{
    int n;
    scanf("%d",&n);
    reverse(n);
}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.