Problem 1: Finding Complexity using Counter Method

Started on	Wednesday, 13 August 2025, 8:17 PM
State	Finished
Completed on	Wednesday, 13 August 2025, 8:23 PM
Time taken	6 mins 22 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

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,i=1,s=1,c=0;
c=c+2;
scanf("%d",&n);
while(s<=n){
c++;
c++;
i++;
c++;
s+=i;
}
c++;
printf("%d",c);
return 0;
}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Finish review

Quiz navigation

1

Finish review

Back to Course

Problem 2: Finding Complexity using Counter method

Started on	Wednesday, 13 August 2025, 8:24 PM
State	Finished
Completed on	Wednesday, 13 August 2025, 8:38 PM
Time taken	13 mins 11 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

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,c=0;
    scanf("%d",&n);
    if(n==1)
        c+=2;
    else{
        c++;
        for(int i=1;i<=n;i++){
            c++;
            for(int j=1;j<=n;j++){
                c+=2;
                c++;
                break;
            }
            c++;
        }
        c++;
    }
    printf("%d",c);
    return 0;
}
```

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

Passed all tests! ✔

Correct

Quiz navigation

1

Finish review

Marks for this submission: 1.00/1.00.

Finish review

Back to Course

Problem 3: Finding Complexity using Counter Method

Started on	Monday, 18 August 2025, 10:30 AM
State	Finished
Completed on	Monday, 18 August 2025, 10:35 AM
Time taken	5 mins 29 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

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 a,count=0;
scanf("%d",&a);
for(int i=1;i<=a;++i){
    count++;
    count++;
    if(a%i==0){
        count++;
}
}
count++;
printf("%d",count);
}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Finish review

Back to Course

Quiz navigation

1

Finish review

Problem 4: Finding Complexity using Counter Method

Started on	Monday, 18 August 2025, 10:37 AM
State	Finished
Completed on	Monday, 18 August 2025, 11:01 AM
Time taken	23 mins 13 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

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 a,c=0,count=0;
    scanf("%d",&a);
    for(int i=a/2;i<a;i++){
        count++;
        count++;
        for(int j=1;j<a;j=2*j){
            count++;
            count++;
            for(int k=1;k<a;k=k*2){
                count++;
                count++;
                c++;
        }}}
    count+=3;
    printf("%d",count-1);
}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Finish review

Back to Course

Problem 5: Finding Complexity using counter method

Started on	Monday, 18 August 2025, 11:10 AM
State	Finished
Completed on	Monday, 18 August 2025, 11:24 AM
Time taken	14 mins 2 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

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,count=0;
    count=count+1;
    scanf("%d",&n);
    while(n!=0){
        count++;
        int r=n%10;
        count++;
        int re=re*10+r;
        count++;
        n/=10;
        count++;
    }
    count++;
printf("%d",count+1);

}
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Finish review

Back to Course