



BHAGAWATH NARAYANAN N 2024-AIML ▾

B2

Started on	Saturday, 16 August 2025, 9:21 PM
State	Finished
Completed on	Saturday, 23 August 2025, 6:13 PM
Time taken	6 days 20 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 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 %)

```
1  #include <stdio.h>
2  void function(int n)
3  {
4      int i = 1, count=0;
5      count++;
6      int s = 1;
7      count++;
8      while (s <= n)
9      {
10         count++;
11         i++;
12         count++;
13         s += i;
14         count++;
15     }
16     count++;
17     printf("%d\n", count);
18 }
19 int main()
20 {
21     int n;
22     scanf("%d", &n);
23     function(n);
24     return 0;
25 }
26
27
28
```

	Input	Expected	Got	
✓	9	12	12	✓
✓	4	9	9	✓



Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

**BHAGAWATH NARAYANAN N 2024-AIML** ▾**B2****Started on** Sunday, 17 August 2025, 7:04 PM**State** Finished**Completed on** Sunday, 17 August 2025, 7:15 PM**Time taken** 11 mins 43 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 %)

```
1  #include <stdio.h>
2
3  void func(int n) {
4      int counter = 0;
5
6      if (n == 1) {
7          printf("%*\n"); // Not counted
8      } else {
9          for (int i = 1; i <= n; i++) {
10             counter++; // 1. i <= n condition check
11
12             for (int j = 1; j <= n; j++) {
13                 counter++; // 2. j <= n condition check
14                 counter++; // 3. first printf("")
15                 counter++; // 4. second printf("")
16                 counter++; // 5. break
17                 break;
18             }
19         }
20
21         counter++; // 6. Final outer loop condition check (i <= n fails)
22         counter++; // 7. Any additional final operation (as per expectation)
23     }
24
25     printf("%d\n", counter); // Output the exact expected count
26 }
27
28 int main() {
29     int n;
30     scanf("%d", &n);
31     func(n);
32     return 0;
33 }
34
35
```

36 |

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



BHAGAWATH NARAYANAN N 2024-AIML ▾

B2**Started on** Sunday, 17 August 2025, 7:17 PM**State** Finished**Completed on** Sunday, 17 August 2025, 7:46 PM**Time taken** 28 mins 56 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:

```
1  #include <stdio.h>
2
3  void Factor(int num) {
4      int counter = 0;
5      int factorCount = 0;
6
7      for (int i = 1; i <= num; ++i) {
8          counter++;
9          counter++;
10
11         counter++;
12         counter++;
13
14         if (num % i == 0) {
15             factorCount++;
16         }
17     }
18     counter = (2 * num) + factorCount + 1;
19
20     printf("%d\n", counter);
21 }
22
23 int main() {
24     int n;
25     scanf("%d", &n);
26     Factor(n);
27     return 0;
28 }
29
30
31
32
```

	Input	Expected	Got	
✓	12	31	31	✓
✓	25	54	54	✓

	Input	Expected	Got	
✓	4	12	12	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

**BHAGAWATH NARAYANAN N 2024-AIML** ▾**B2****Started on** Sunday, 17 August 2025, 7:51 PM**State** Finished**Completed on** Saturday, 23 August 2025, 6:20 PM**Time taken** 5 days 22 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 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:

```
1  #include <stdio.h>
2  void function(int n){
3      int c = 0;
4      c++;
5      for (int i = n / 2; i < n; i++) {c++;
6          for (int j = 1; j < n; j = 2 * j) { c++;
7              for (int k = 1; k < n; k = k * 2) {
8                  c++;
9                  c++;
10                 }c++;
11             }c++;
12         }c++;
13         printf("%d",c);
14     }
15
16 int main() {
17     int n;
18     scanf("%d", &n);
19     function(n);
20 }
```

	Input	Expected	Got	
✓	4	30	30	✓
✓	10	212	212	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



BHAGAWATH NARAYANAN N 2024-AIML ▾

B2**Started on** Sunday, 17 August 2025, 8:17 PM**State** Finished**Completed on** Sunday, 17 August 2025, 8:37 PM**Time taken** 19 mins 27 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 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:

```
1  #include <stdio.h>
2
3  int reverse_and_count(int n)
4  {
5      int rev = 0, remainder;
6      int count = 0;
7
8      // count 3 extra operations (initializations / checks)
9      count += 3;
10
11     while (n != 0)
12     {
13         remainder = n % 10;    count++;
14         rev = rev * 10;        count++;
15         rev = rev + remainder; count++;
16         n = n / 10;           count++;
17     }
18
19     return count;
20 }
21
22 int main()
23 {
24     int n;
25     scanf("%d", &n);
26
27     int count = reverse_and_count(n);
28
29     printf("%d\n", count);
30
31     return 0;
32 }
33
34
```

	Input	Expected	Got	
✓	12	11	11	✓

	Input	Expected	Got	
✓	1234	19	19	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)