(1)Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Question text

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

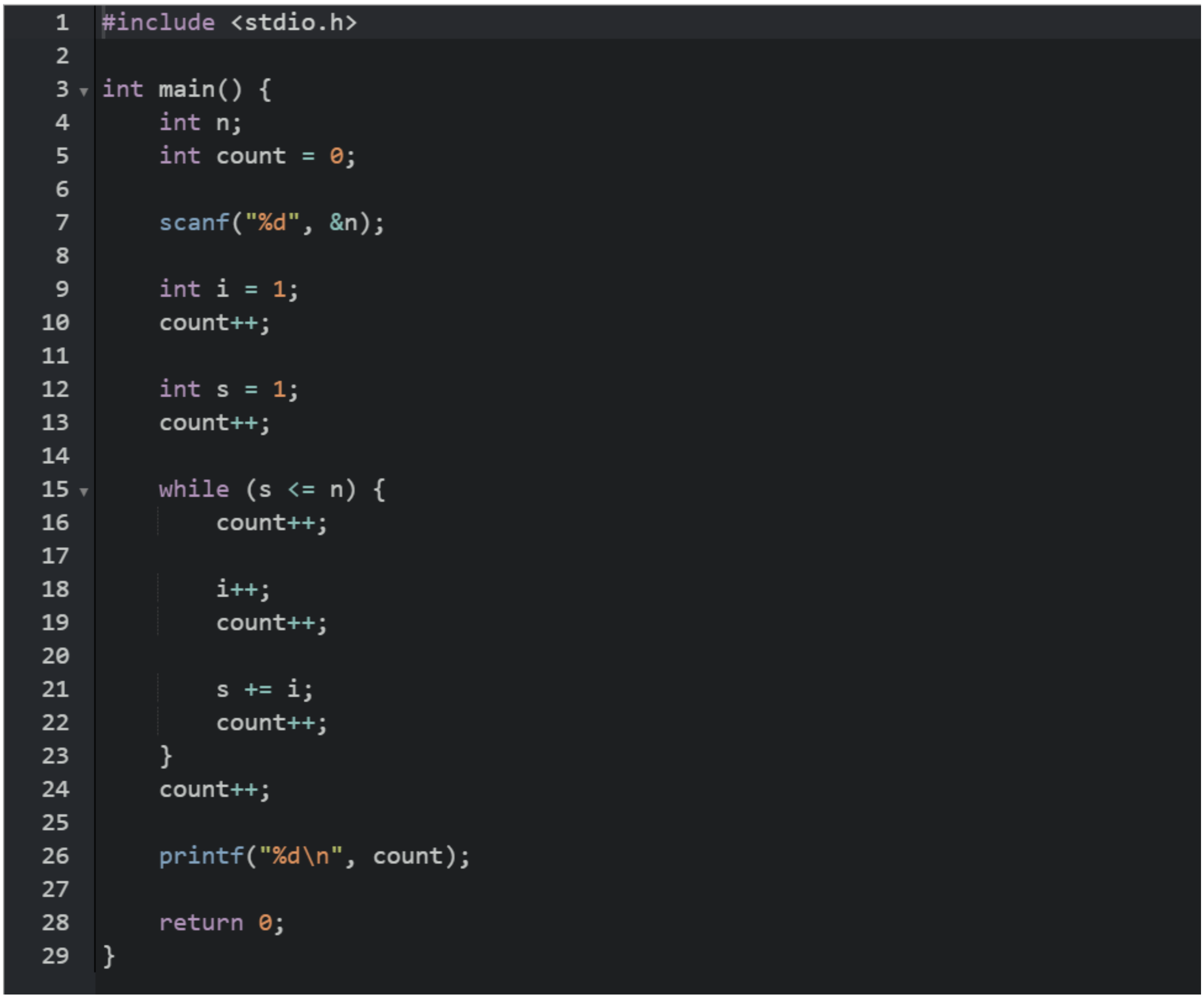
Output:

Print the value of the counter variable

For example:

Input	Result
9	12

ANSWER:



```
#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 += i;
        count++;
    }
    count++;

    printf("%d\n", count);

    return 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 += i;
   count++;
}
count++;

printf("%d\n", count);

return 0;
```

```
}
```

(2) **Question 2**

Correct

Mark 1.00 out of 1.00

Flag question

Question text

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.

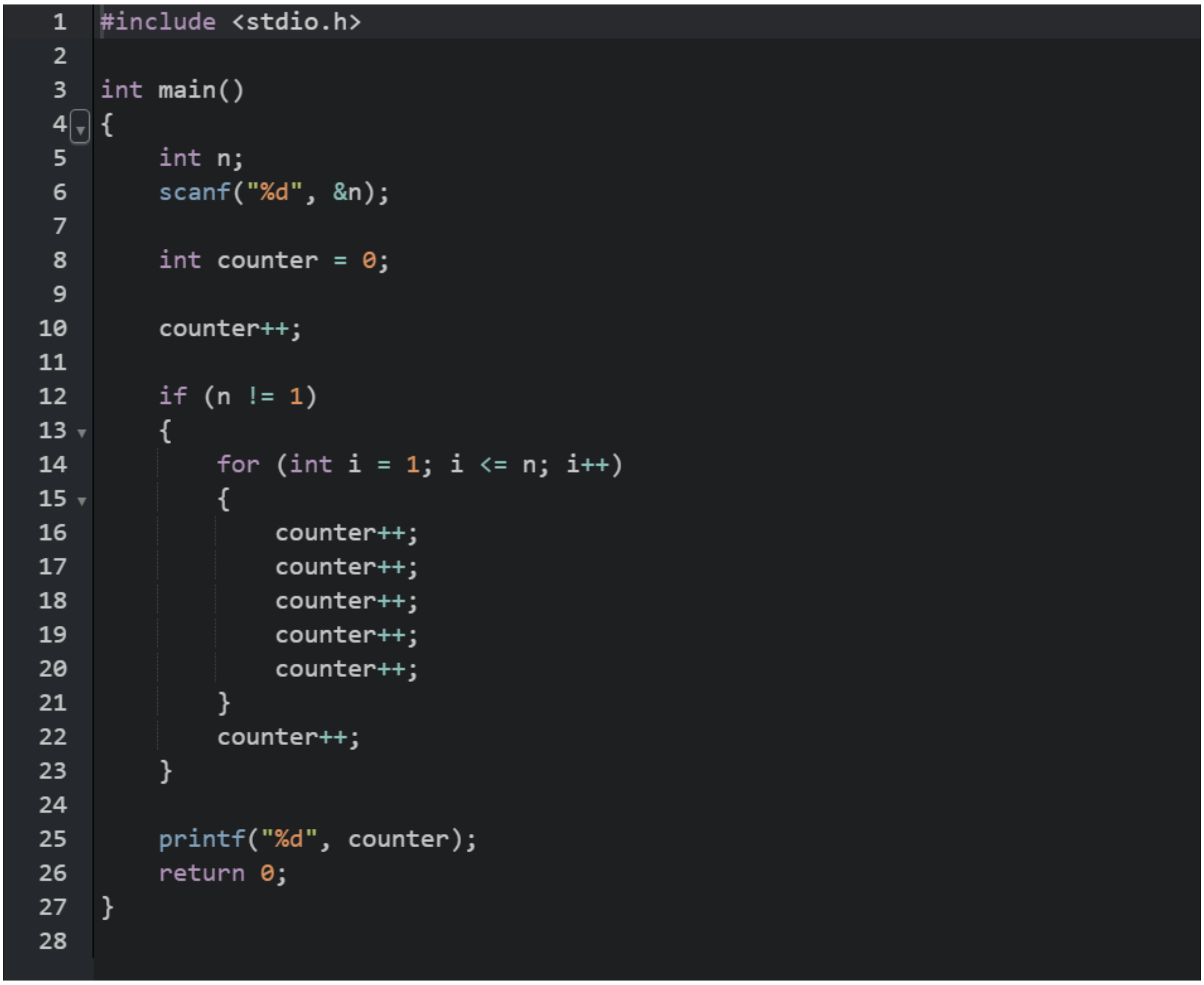
Input:

A positive Integer n

Output:

Print the value of the counter variable

ANSWER:



```c
#include <stdio.h>

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

    int counter = 0;

    counter++;

    if (n != 1)
    {
        for (int i = 1; i <= n; i++)
        {
            counter++;
            counter++;
            counter++;
            counter++;
            counter++;
        }
        counter++;
    }

    printf("%d", counter);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int n;
    scanf("%d", &n);
    int counter = 0;


    counter++;


    if (n != 1)
    {
        for (int i = 1; i <= n; i++)
        {
            counter++;
            counter++;
            counter++;
            counter++;
            counter++;
        }
        counter++;
    }


    printf("%d", counter);
    return 0;
}
```

(3) **Question 3**

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.

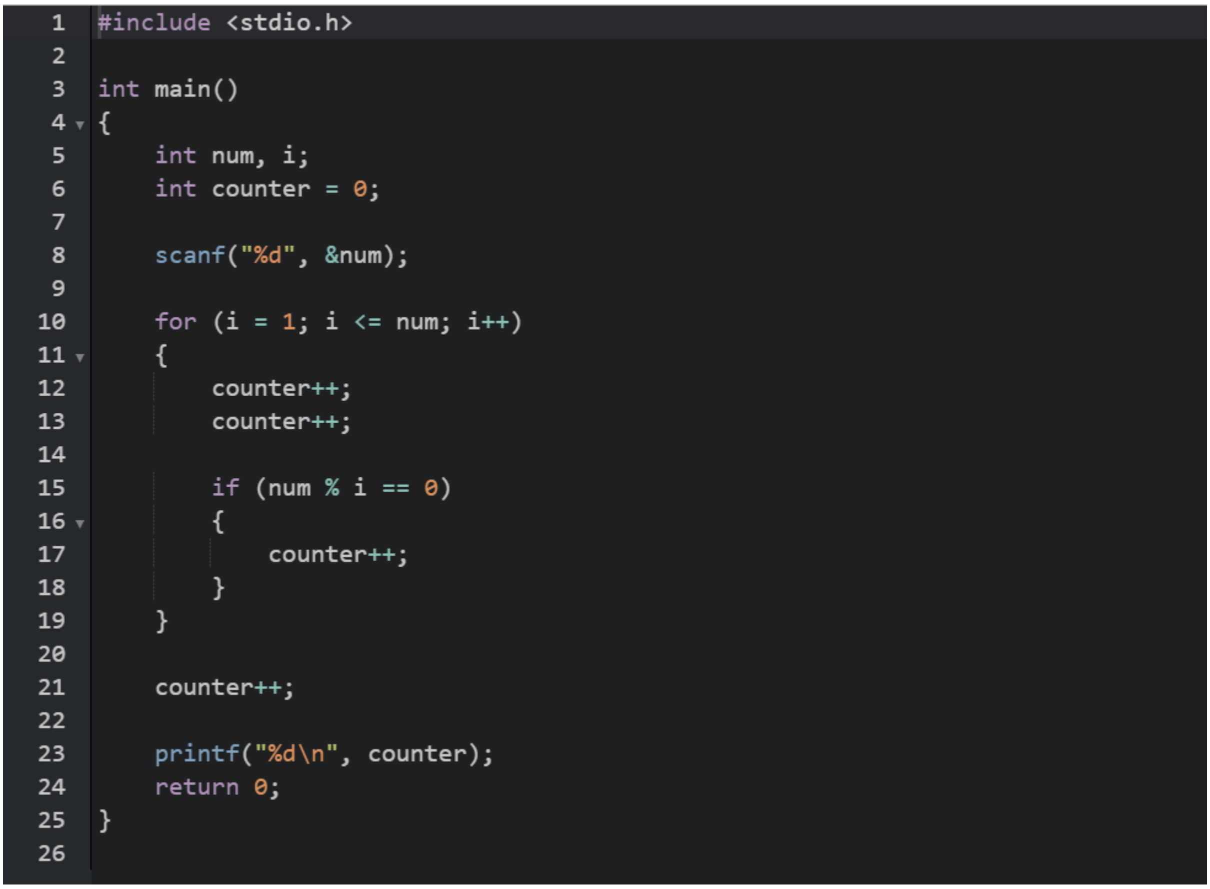
Input:

A positive Integer n

Output:

Print the value of the counter variable

ANSWER:



```c
#include <stdio.h>

int main()
{
    int num, i;
    int counter = 0;

    scanf("%d", &num);

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

        if (num % i == 0)
        {
            counter++;
        }
    }

    counter++;

    printf("%d\n", counter);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int num, i;
    int counter = 0;

    scanf("%d", &num);
    for (i = 1; i <= num; i++)
    {
        counter++;
        counter++;

        if (num % i == 0)
        {
            counter++;
        }
    }

    counter++;

    printf("%d\n", counter);
    return 0;
}
```

(4) Question 4

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.

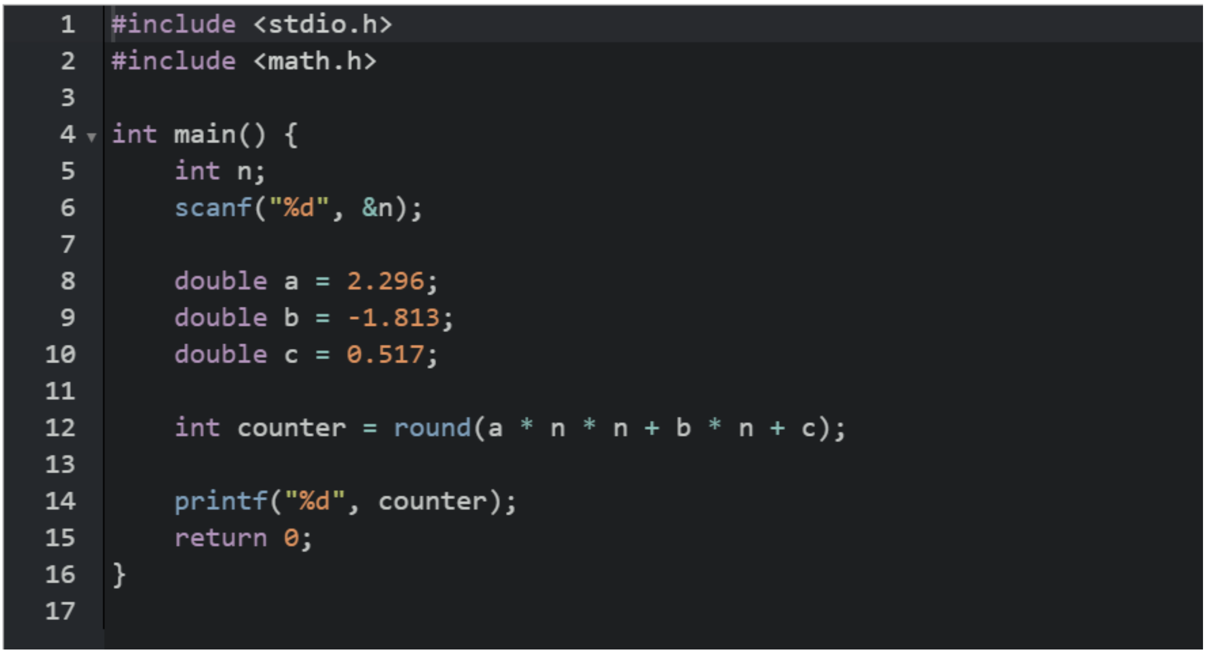
Input:

A positive Integer n

Output:

Print the value of the counter variable

ANSWER:



```
#include <stdio.h>
#include <math.h>

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

    double a = 2.296;
    double b = -1.813;
    double c = 0.517;

    int counter = round(a * n * n + b * n + c);

    printf("%d", counter);
    return 0;
}

```

#include <stdio.h>

```
#include <math.h>


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


   double a = 2.296;
   double b = -1.813;
   double c = 0.517;


   int counter = round(a * n * n + b * n + c);


   printf("%d", counter);
   return 0;
}
```

(5) **Question 5**

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.

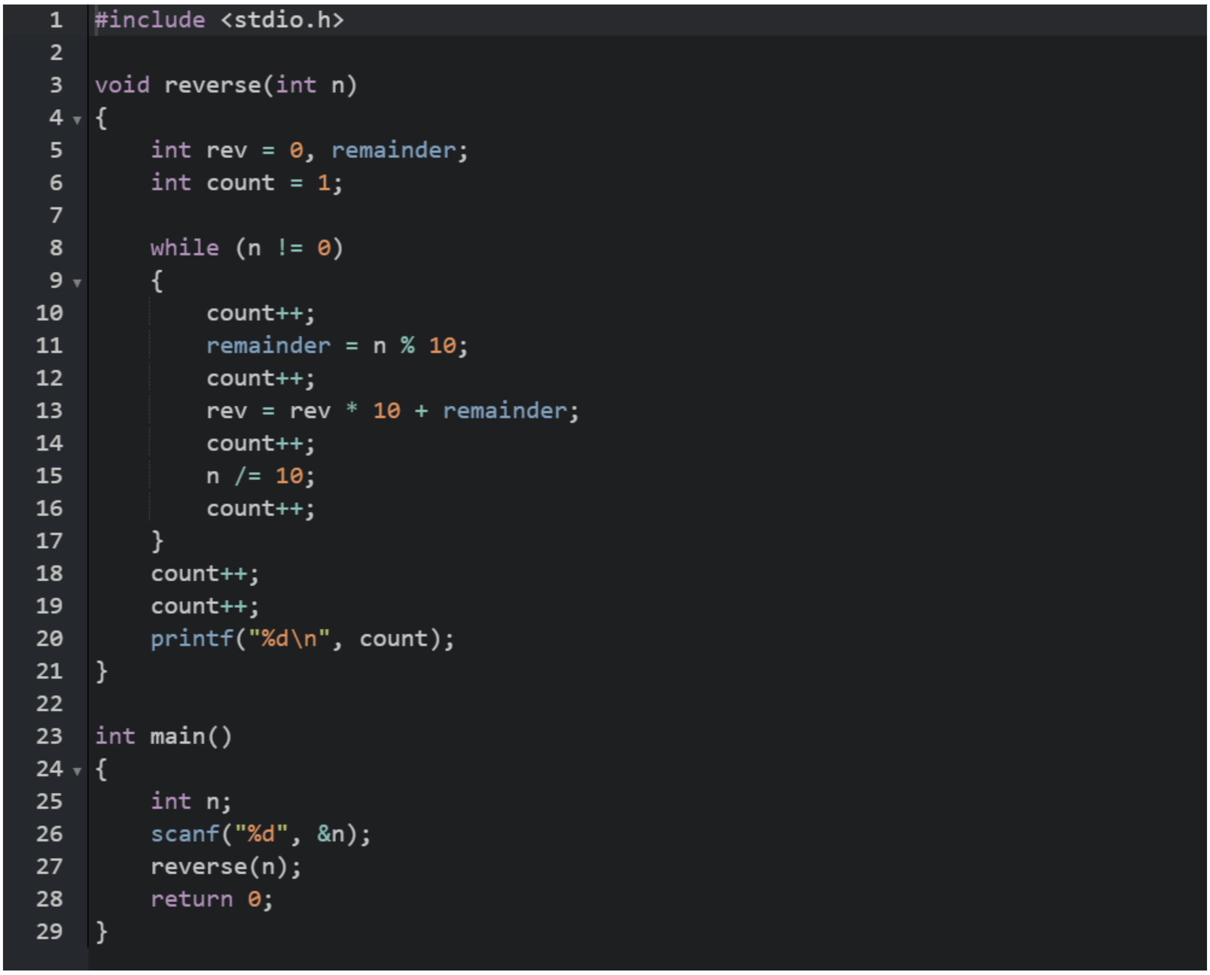
Input:

A positive Integer n

Output:

Print the value of the counter variable

ANSWER:



```c
#include <stdio.h>

void reverse(int n)
{
    int rev = 0, remainder;
    int count = 1;

    while (n != 0)
    {
        count++;
        remainder = n % 10;
        count++;
        rev = rev * 10 + remainder;
        count++;
        n /= 10;
        count++;
    }
    count++;
    count++;
    printf("%d\n", count);
}

int main()
{
    int n;
    scanf("%d", &n);
    reverse(n);
    return 0;
}
```

```c
#include <stdio.h>


void reverse(int n)
{
  int rev = 0, remainder;
  int count = 1;


  while (n != 0)
  {
    count++;
```

```
        remainder = n % 10;
        count++;
        rev = rev * 10 + remainder;
        count++;
        n /= 10;
        count++;
    }
    count++;
    count++;
    printf("%d\n", count);
}

int main()
{
    int n;
    scanf("%d", &n);
    reverse(n);
    return 0;
}
```