Solution Review 2: Count Iterations of a Loop Until a Condition

This lesson gives a detailed solution review of the challenge in the previous lesson.

We'll cover the following

- Solution:
  - Explanation

Solution:

```
fn test(mut x:i32) {
    // define a mutable variable
    let mut count = 0;
    // define a while loop
    while x >= 0 {
       x = x - 3; // decrement the value of x by 3
       count = count + 1;
    }
    println!("{}", count);
}
fn main(){
    print!("Iterations when x = 21 :");
    test(21);
    print!("Iterations when x = 33 :");
    test(33);
}
```

Explanation

- On **line 3**, a mutable variable `count` is initialized with `0`.
- `while` **construct**
  - `while` definition
    - On **line 5**, `while` condition `x >= 0` is defined i.e., the loop terminates when `x` becomes negative.
  - `while` body

- `while` body

  The body of `while` loop is defined from **line 5** to **line 8**.

  - On **line 6**, a variable `x` is decremented by 3.
  - On **line** 7, the value of `count` is incremented each time within the loop.

- When the loop breaks, the value of `count` is printed which gives the total number of iterations of the `while` loop.

The following illustration explains the code above.

> Assume that `x` is assigned the value 24 and passed to the function.

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

21


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

21

0

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

21 x >= 0 => True

0


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

18

0


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

18

1

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

18 x >= 0 => True

1


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

15

1


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

15

2

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

15 x >= 0 => True

2


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

12

2


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

12

3

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

12 x >= 0 => True

3


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

9

3


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

9

4

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

9 `x >= 0 => True`

4


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

6

4


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

6

5

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

6 **x >= 0 => True**

5


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

3

5


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

3

6

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

3 **x >= 0 => True**

6


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

0

6


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

0

7

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

0 **x >= 0 => True**

7


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

-3

7


```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
        x = x - 3;
        count = count + 1;
    }
    println!("{}", count);
}
```

-3

8

```
fn test(mut x:i32) {
    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

-3 **x >= 0 => False**

8


```
fn test(mut x:i32) {

    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}
```

Output:

8


```
fn test(mut x:i32) {

    let mut count = 0;
    while x >= 0 {
       x = x - 3;
       count = count + 1;
    }
    println!("{}", count);
}end of program code
```

Output:

8