

# do while Loops

<https://csci-1301.github.io/about#authors>

October 27, 2021 (03:18:49 PM)

## Contents

<b>1 Do while Loops</b>	<b>1</b>
1.1 Implementing do while Loops . . . . .	1
<b>2 while vs do while</b>	<b>2</b>

## 1 Do while Loops

Before writing code, think through the following problems:

- In your own words, what is the difference between `while` and `do while` loops?
- Can you think of a problem where
  - using `while` is preferable over `do while` loop?
  - using `do while` is preferable over `while` loop?

### 1.1 Implementing do while Loops

In all the problems in this section, use a `do while` loop.

1. Write a program that display numbers 0 to 50.
2. Write a program that display numbers 30 to -20.
3. Write a `do while` loop that generates this output:

```
1 10 100 1000 10000 100000 1000000
```

In the next problem, implement a program combining `do-while` loop with user input to achieve the following behavior:

1. Ask user to enter an integer in the range 0 - 100 (including 0 and 100).
2. If the value provided by user is not in this range, the program should repeat the question.
3. After use provides an integer within the range, display that number.

Here is an example of possible program output:

Enter an integer between 0 and 100: NO  
Sorry, that is not a valid input.

Enter an integer between 0 and 100: -20  
Sorry, that is not a valid input.

Enter an integer between 0 and 100: 42  
You entered 42.

## 2 while vs do while

Consider the program given below implemented using a **while** loop:

```
int n;  
bool flag = false;  
  
Console.Write("Enter an integer:");  
flag = int.TryParse(Console.ReadLine(), out n);  
  
while(!flag)  
{  
    Console.WriteLine("The value you entered is not a valid integer. Try one more  
    ↪ time.");  
    Console.Write("Enter an integer:");  
    flag = int.TryParse(Console.ReadLine(), out n);  
}  
  
Console.WriteLine($"The number you entered is {n}");
```

1. Convert the program to an equivalent version that uses a **do while** loop
2. Test the **do while** version with different inputs to ensure the behavior is the same and that the program does not crash with error. For example, you should try:
  - alphabetic input (invalid)
  - floating point input (invalid)
  - negative integer (valid)
  - positive integer (valid)
  - number 0 (valid)
3. Compare the **while** and **do while** implementations: which one is better, in your opinion, and why?