while and do while

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

June 16, 2021 (09:32:08 PM)

Contents

1 Practicing while Loops

1.1 Problem 1

- Create a new project, and replace the content of the *main* method with the following code

```
int i = 0;
while(i<100)
{
    Console.WriteLine(i);
    i++;
}
```

- Execute the code. You should see the numbers 0 to 99 in the console.
- Without changing the numbers, modify the code such that it prints 0 to 100 in the console. Note the diffrence between < and <= operators.
- Modify the code such that it prints the numbers from 100 to 300. Note that the counter can start from any number you wish.

- Modify the code such that it prints all integers between 0 and 100 that are divisible by 3.
- To implement the above problem, you may code one of the following:

```
int i = 0;
while(i<100)
{
    if(i%3 ==0)
        Console.WriteLine(i);
    i++;
}
```

or

```
int i = 0;
while(i<100)
{
    Console.WriteLine(i);
    i+=3;
}
```

- Which one of the above codes is more efficient? Why?
- Note that you do not have to increment the counter only by one each time. You should update the counter wisely and try to use it more efficiently.

1.2 Problem 2

- Create a new project and replace the content of the *main* method with the following code:

```
int n;
Console.Write("Enter a natural number greater than 2: ");
n = int.Parse(Console.ReadLine());
int i = 2;
while(n % i != 0 && i < n )
{
    i++;
}
if (i == n)
    Console.WriteLine($"{n} is a ... number");
else
    Console.WriteLine($"{n} is not a ... number");
```

- What does the code do? Explain the boolean expession of the looop
- Replace ... with a meaningful word.

1.3 Problem 3

- Create a new project and replace the content of the *main* method with the following code:

```
int n = 100;
while (n > 0 )
{
    Console.WriteLine(n);
    n--;
}
```

- Execute the code, and explain what you see in the console. Note that the counter is incremental.

1.4 Problem 4

Write a program that gets a number from the console and finds its biggest divisor less than the number itself.

1.5 Problem 5

Write a new program that asks an integer value greater than 1 from the user, and computes the result of this series: `1 + 2 + 3 + 4 + ...` up to `n` where `n` represents the number obtained from the user.

1.6 Problem 6

Ask user to enter integers. Keep track of the smallest value user enters. After user indicates they are done, display the smallest value user entered. If user did not enter any integers display `You did not enter anything`.

2 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 using **`while`** preferred over the **`do while`** loop? In all the problems in this section, use **`do while`** loop.

2.1 Problem 1

Write a program that display numbers 0 to 50.

2.2 Problem 2

Write a program that display numbers 30 to -20.

2.3 Problem 3

Write a **`do while`** loop that generates this output: `1 10 100 1000 10000 100000 1000000`

3 `while` vs `do while`

Both of the following programs keep getting inputs from the user until the user enters a valid integer. Which one is better? Explain your answer.

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

Console.WriteLine($"The number you entered is {n}");
```

```
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}");
```

4 Infinite Loops

All of the following are examples of infinite loops. Can you spot the problem? How would you change the code to fix it?

```
int number = 0;
while (number <=5) {
    Console.WriteLine("Hi!");
    Console.WriteLine(number);
}
```

```
int number1 = 0, number = 0;
while (number <=5) {
    Console.WriteLine("Hi!");
    Console.WriteLine(number);
    number1++;
}
```

```
int number = 0;
while (number <=5);
{
    Console.WriteLine("Hi!");
    Console.WriteLine(number);
    number++;
}
```

```
int number = 0;
while (number <=5)
Console.WriteLine("Hi!");
Console.WriteLine(number);
number++;
```

```
int number = 0;
while (number <= 5)
{
    Console.WriteLine("Hi!");
    Console.WriteLine(number);
}
number++;
```

5 ++ and -- Operators

For each of the following pieces of code, determine the final value of n. Explain your answer.

```
int x = 5;
int n = x++;
```

```
int x = 5;
int n = ++x;
```

```
int x = 5;
int n = x++ + x++;
```

```
int x = 5;
int n = ++x + ++x;
```

```
int x = 5, y = 6;
int n = x++ * ++y;
```

```
int x = 5;
int n = x++ + --x;
```