

# INF 154 PRACTICAL 6 2023



UNIVERSITEIT VAN PRETORIA  
UNIVERSITY OF PRETORIA  
YUNIBESITHI YA PRETORIA

Denkleiers • Leading Minds • Dikgopolo tša Dihalefi

# At the end this practical you should be able...

1. To work with while loops
2. To avoid infinite while loops
3. To use while loops to solve common problems, such as, finding the sum, average, minimum, maximum and sorting numbers, etc.
4. To work with nested while loops



## Hints for Practical 6

***Make sure you understand the counter-controlled while loop examples used in Lecture 6 to display***

- Numbers from 1-10 or Numbers from 99 to 0
- Odd numbers or Even Numbers
- The sum of all the looped numbers

***Make sure you understand the sentinel controlled while loop examples used in Lecture 6 to demonstrate***

- The index position of an item in a list box
- The number of times a program takes to guess a defined variable.

***For this practical session, you should be able to use***

- The different windows form toolbox controls (L1)
- Input-process-output programming model (L2 and L3)
- Variables (L2 and L3)
- Decision statements (L4 and L5)
- Random number generators (L5)



# What are we going to do?

- Practical 6a: Completed in class.
- Practical 6b: Completed in class.
- Practical 6c: Take home practical.

# Practical Exercise 6a

**Objective:** In this practical section, you are tasked with creating a windows form program which will change a gender into a string or character format. The program should also be able to display the count of each gender. This should be archived using **While Loops & Decision Statements**.

The conversion formats are as follows

“Male” to “M”

“Female” to “F”

“M” to “Male”

“F” to “Female”

# Practical Exercise 6a

## Expected Screen Design

Create the following form with the appropriate controls. When the form loads, all the genders should be in the same format. Remember to name your controls accordingly (e.g for a button you would use btnGenderStats).

Listbox

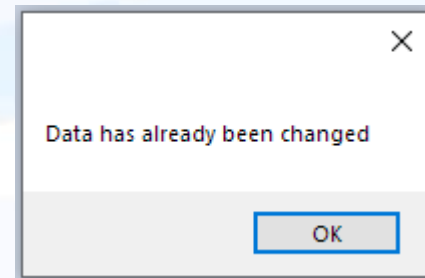
Button  
Controls

The screenshot shows a Windows application window titled "Gender Formats". Inside the window, there is a listbox containing the following text entries: M, M, M, F, M, F, F. Below the listbox, there are three buttons stacked vertically: "To String", "To Character", and "Gender Statistics".

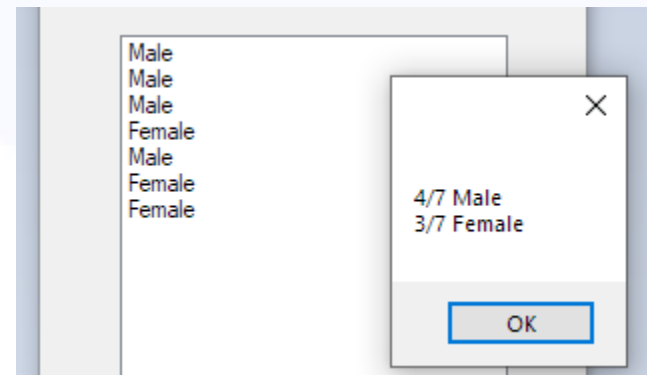
# Practical Exercise 6a

## Expected Functionality

- When the To String button is clicked, All the gender characters should change from characters to string formats.
- When the To Character button is clicked, All the gender strings should change from string to character.
- When same button is clicked multiple times in succession, A single message box should display as follows.



- When the display button is clicked, A message box showing the male and female count should be shown.



# Practical Exercise 6a

## Solution - btnToString

```
1 reference
private void btnToString_Click(object sender, EventArgs e)
{
    // string to character
    int i = 0;
    while (i < lstGenders.Items.Count)
    {
        if (lstGenders.Items[i].ToString() == "M")
        {
            lstGenders.Items[i] = "Male";
        }
        else if (lstGenders.Items[i].ToString() == "F")
        {
            lstGenders.Items[i] = "Female";
        }
        else
        {
            MessageBox.Show("Data has already been changed");
            break;
        }
        i++;
    }
}
```





# Practical Exercise 6a

## Solution - btnToCharacter

```
private void btnToCharacter_Click(object sender, EventArgs e)
{
    // character to string
    int i = 0;
    while (i < lstGenders.Items.Count)
    {
        if (lstGenders.Items[i].ToString() == "Male")
        {
            lstGenders.Items[i] = 'M';
        }
        else if (lstGenders.Items[i].ToString() == "Female")
        {
            lstGenders.Items[i] = "F";
        }
        else
        {
            MessageBox.Show("Data has already been changed");
            break;
        }
        i++;
    }
}
```



# Practical Exercise 6a

## Solution - btnGenderStats

```
1 reference
private void btnGenderStats_Click(object sender, EventArgs e)
{
    int i = 0;
    int maleCounter = 0, femaleCounter = 0;
    while (i < lstGenders.Items.Count)
    {
        if (lstGenders.Items[i].ToString() == "Male" || lstGenders.Items[i].ToString() == "M")
        {
            maleCounter++;
        }
        else
        {
            femaleCounter++;
        }
        i++;
    }
    MessageBox.Show( maleCounter + "/" + i + " Male" + "\n" + femaleCounter + "/" + i + " Female" );
}
```



# Practical Exercise 6b

**Objective:** In this practical section, you are tasked with creating a windows form program which will print out a  $n^{\text{th}}$ -by- $n^{\text{th}}$  multiplication table. Ensure proper space placement for readability. This should be archived using **Nested while Loops & a Random Generator.**



# Practical Exercise 6b

## Expected Screen Design

Create the following form with the appropriate controls. When the Generate Table button is clicked: It should appear as follow: Remember to name your controls accordingly (e.g for a button you would use btnGenerate).

Richtextbox

Button

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81

Generate Table

# Practical Exercise 6b

## Solution - btnGenerator

```
1 reference
private void btnGenerate_Click(object sender, EventArgs e)
{
    Random MyRandom = new Random();
    int n = MyRandom.Next(1,11);

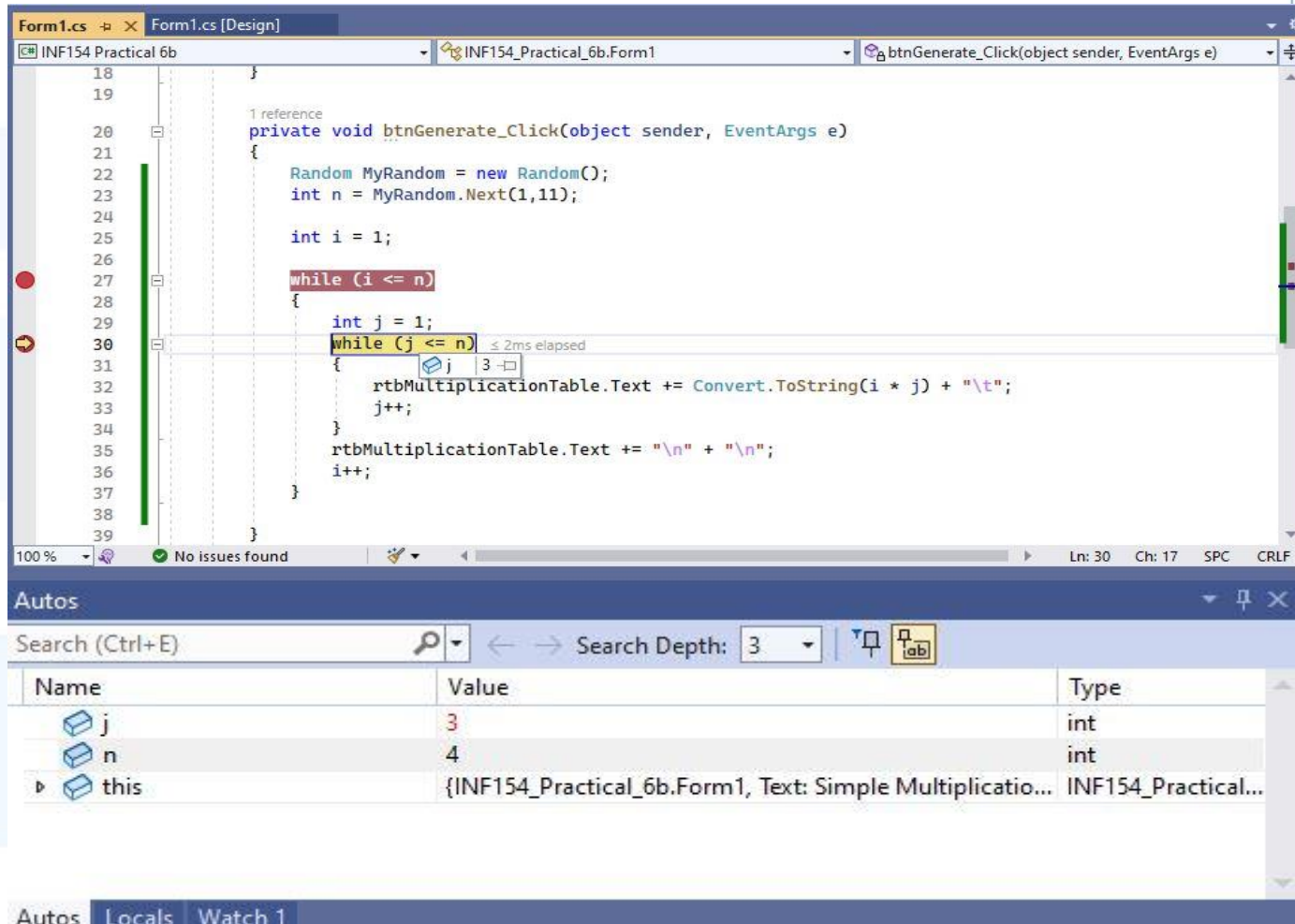
    int i = 1;

    while (i <= n)
    {
        int j = 1;
        while (j <= n)
        {
            rtbMultiplicationTable.Text += Convert.ToString(i * j) + "\t";
            j++;
        }
        rtbMultiplicationTable.Text += "\n" + "\n";
        i++;
    }
}
```



# Practical Exercise 6b

**Make use of breakpoints to test and evaluate your code:**



The screenshot shows the Visual Studio IDE with a C# file named `Form1.cs` open. The code is for a button click event handler `btnGenerate_Click`. A breakpoint is set on the line `while (j <= n)` at line 30. The code is as follows:

```
18 }
19
20 1 reference
21 private void btnGenerate_Click(object sender, EventArgs e)
22 {
23     Random MyRandom = new Random();
24     int n = MyRandom.Next(1,11);
25
26     int i = 1;
27     while (i <= n)
28     {
29         int j = 1;
30         while (j <= n)
31         {
32             rtbMultiplicationTable.Text += Convert.ToString(i * j) + "\t";
33             j++;
34         }
35         rtbMultiplicationTable.Text += "\n" + "\n";
36         i++;
37     }
38 }
39
```

The Autos window is open, showing the current state of variables:

Name	Value	Type
j	3	int
n	4	int
this	{INF154_Practical_6b.Form1, Text: Simple Multiplicatio...	INF154_Practical...

# Practical Exercise 6c

**Objective:** In this practical section, there will be two parts. For part one-- you are tasked with creating a program that will generate random grades between 0-100. This program should be able to do the following:

- to calculate the average of all of its grades.
- to tell you how many fall into the pass/fail category.
- to sort the grades in ascending order.

# Practical Exercise 6c

## Part One

### Expected Screen Design

Create the following form with the appropriate controls. When the form loads, all the genders should be in the same format. Remember to name your controls accordingly (e.g for a button you would use btnSubmit).

The screenshot shows a Windows application window titled "Practical 6". Inside the window, there is a form with two tabs: "Part ONE" and "Part TWO". The "Part ONE" tab is active. On this tab, there is a "Generate Grades" button at the top. Below it is a listbox labeled "lstGrades". To the right of the listbox is a group box labeled "Options:" containing three radio buttons: "Average Grade", "Categorize Grades (P/F)", and "Sort Grades (Ascending)". At the bottom of the form is a "Submit" button. An orange box with the text "Listbox" and an arrow points to the "lstGrades" listbox.



# Practical Exercise 6c

## Expected Functionality

1. By clicking the button Generate Grades, the program should populate the list box with 10 random grades with one decimal point.

Practical 6

Part ONE Part TWO

Generate Grades

21,3  
90,5  
80,7  
80,6  
86,7  
17,6  
56,4  
68,1  
70,7  
2,7

Options:

☐ Average Grade

☐ Categorize Grades (P/F)

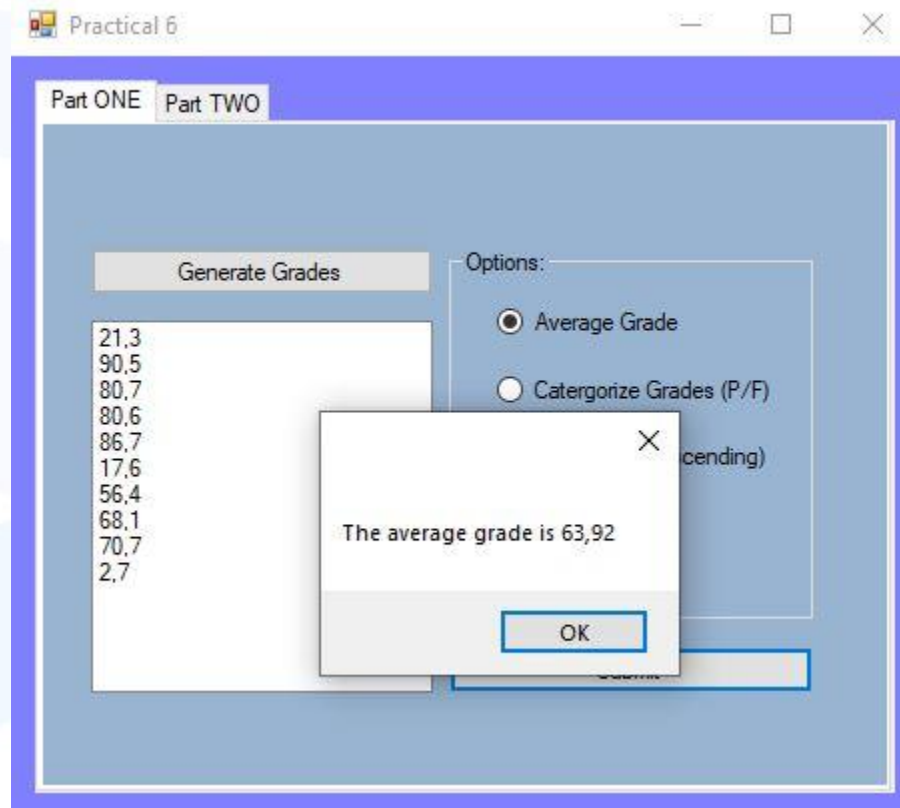
☐ Sort Grades (Ascending)

Submit



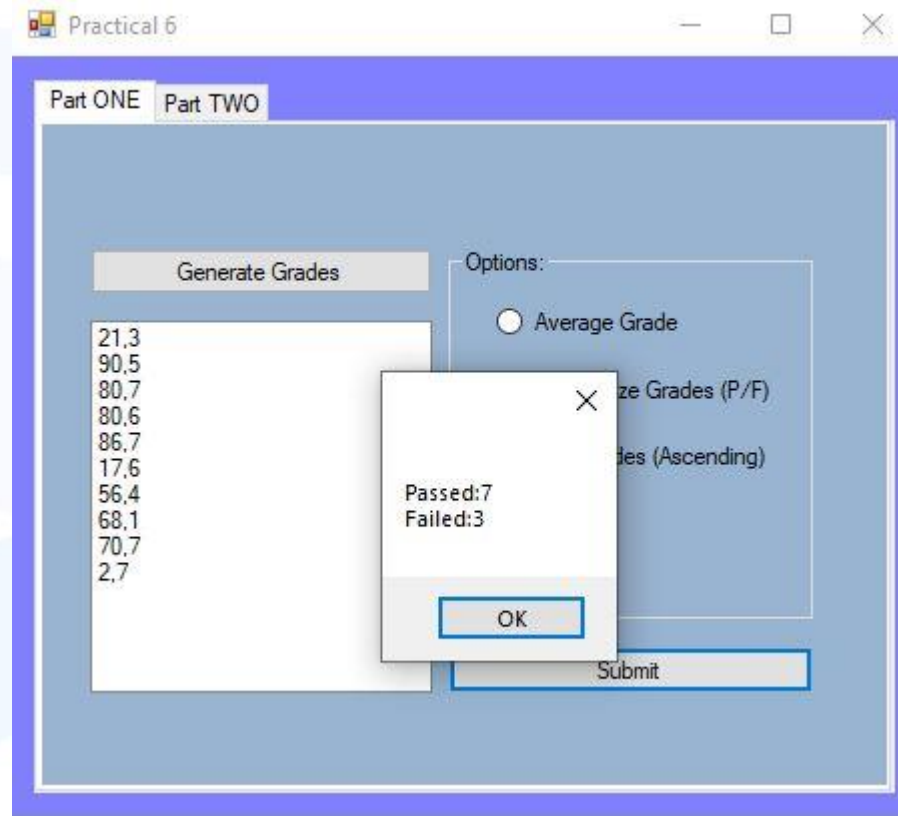
# Practical Exercise 6c

2. By selecting the average grade radio button and clicking the submit button, the program should calculate the average grade.



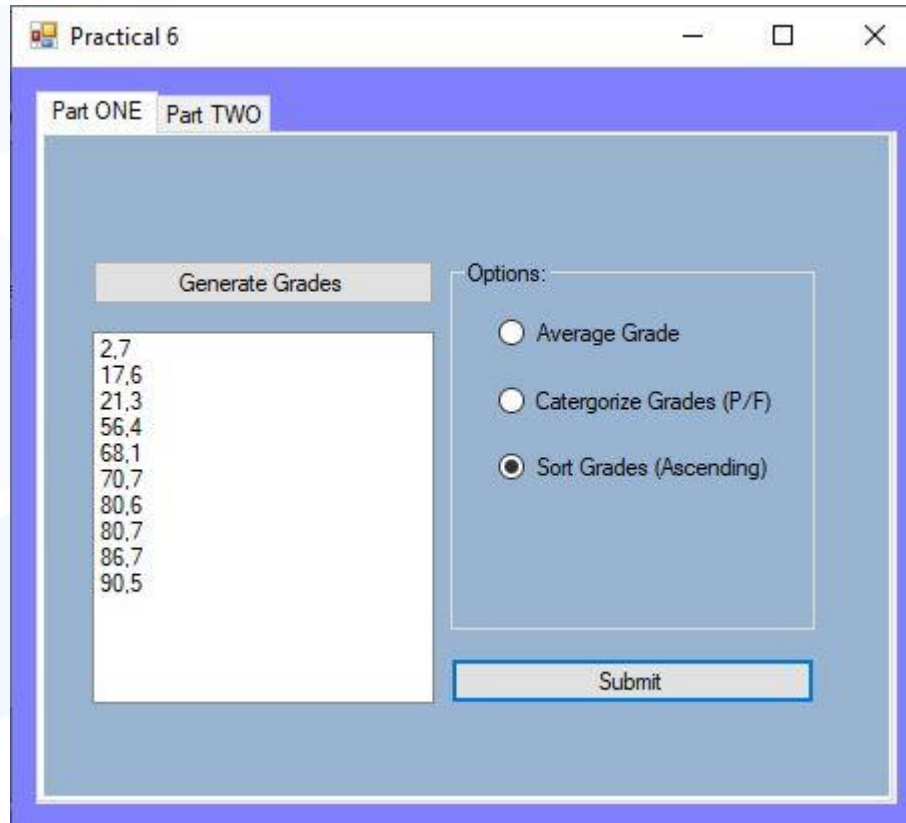
# Practical Exercise 6c

3. By selecting the categories grade radio button and clicking the submit button, the program should display the counts of passes and fails.



# Practical Exercise 6c

4. By selecting the sort grades radio button and clicking the submit button, the program should sort the grades in ascending order.



The screenshot shows a Java Swing window titled "Practical 6" with a blue border. Inside the window, there are two tabs: "Part ONE" and "Part TWO". The "Part ONE" tab is active. It contains a "Generate Grades" button at the top. Below this button is a text area displaying a list of grades: 2,7, 17,6, 21,3, 56,4, 68,1, 70,7, 80,6, 80,7, 86,7, and 90,5. To the right of the text area is an "Options:" section with three radio buttons: "Average Grade", "Categorize Grades (P/F)", and "Sort Grades (Ascending)". The "Sort Grades (Ascending)" radio button is selected. Below the "Options:" section is a "Submit" button.

Grades
2,7
17,6
21,3
56,4
68,1
70,7
80,6
80,7
86,7
90,5

# Practical Exercise 6c

## Part Two

This is how Part two's interface should look. You will need to create a program that will allow the user to enter a value between 1 and 10 and calculate the factorial of the inputted number and enter that value in the textbox.

Practical 6

Part ONE Part TWO

While Loop Factorial Exercise:

Enter a number between 1 and 10: 1

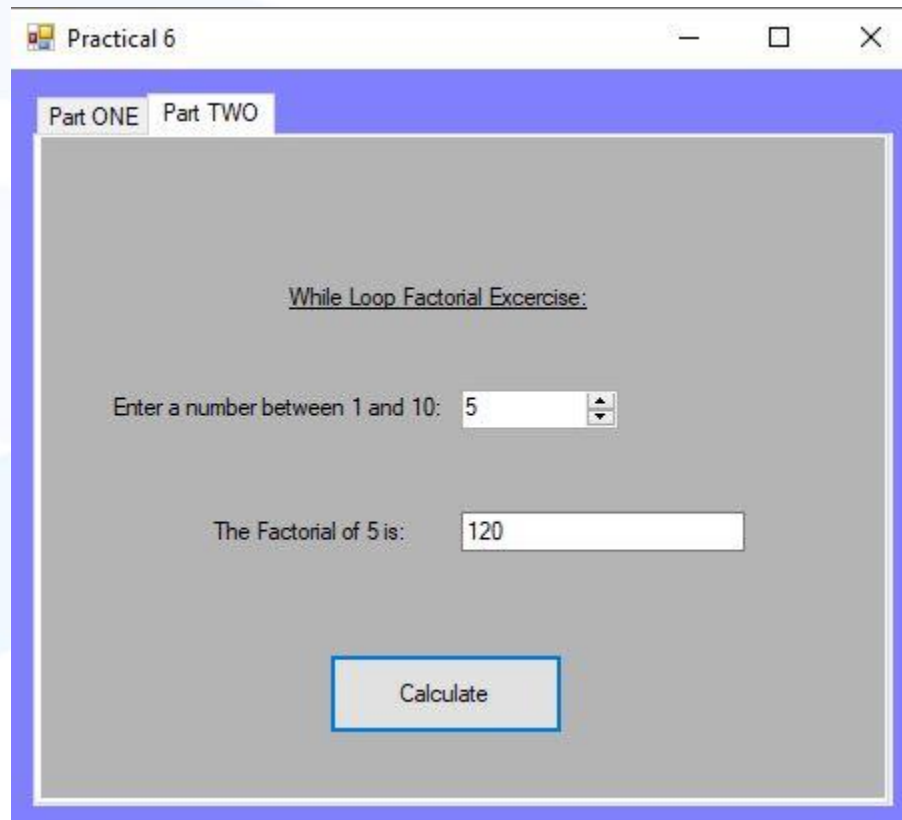
The Factorial of \_\_\_\_ is:

Calculate

# Practical Exercise 6c

## Part Two

In this example, the user entered 5... which meant that the calculation was  
 $5 \times 4 \times 3 \times 2 \times 1 = 120$



The screenshot shows a Windows application window titled "Practical 6". Inside the window, there are two tabs: "Part ONE" and "Part TWO", with "Part TWO" currently selected. The main content area has a grey background and contains the following elements:

- The text "While Loop Factorial Exercise:" centered.
- A label "Enter a number between 1 and 10:" followed by a text box containing the number "5".
- A label "The Factorial of 5 is:" followed by a text box containing the number "120".
- A blue button labeled "Calculate" at the bottom center.



# Practical 6c Submission

Submit your Practical 6c project on ClickUP as follows:

- **Due Date: 1 May 2023.**
- Name your project, **\*student number\* Practical 6** and compress (zip folder) your project.
- Submit it under the Practical 6 submission link.

