

In Class 6 – **while** Loops

In this assignment, you will be writing a C++ program in which you write a bunch of `while` loops!

Goals

Practice with

- writing lots of `while` loops

Problem Statement

You are writing 4 individual loops for this assignment. The last loop is a "Do you want to do that again?" type of loop around the next to last loop. Follow the instructions below.

1. Write a snippet of code that prompts the user for a number and adds up the even numbers from 1 to that entered number.

```
Enter the number to add the even numbers up to: 33
LOOP1 total = 272
```

2. Write another snippet of code that prompts the user for 2 numbers and adds up the numbers between (and including) the numbers.

```
Enter a lower and upper number to sum: 22 44
LOOP2 total = 759
```

3. Write another snippet of code that prompts the user for file name and adds up all of the numbers in the file. Read until EOF (end of file).

```
Enter a file name to read from: in.in
LOOP3 total = 18.0000
```

4. Write another snippet of code **around the code for #3** that prompts the user whether she or he wants to prompt and read another file.

```
Enter a file name to read from: in.in
LOOP3 total = 18.0000
Do you want to do all of this again? y
```

When the user enters **n**, then the program continues on to Part #5.

```
Enter a file name to read from: in
error opening file!
```

Error message when the file does not open (does not exist).

5. Write another snippet of code that prompts the user for file name and reads & adds up the first 5 numbers in the file. (Part 5 executes even when the error message is printed in Part4.)

```
Enter another file name to add 5 numbers from: in.in
LOOP4 total = 18.0000
```

6. Finally, add a snippet of code to print the 3rd number in the file.

```
Enter another file name to add 5 numbers from: in.in
number3 = -1.0000
LOOP4 total = 18.0000
```

Input/Output Example

You must create your own input files so you can determine the output. For these tests, I am only using files with 5 or 7 numbers in them.

Your output should look like this: (Please copy and paste the output text AND prompts into your code! I match on this text to grade your programs. Use fixed and setprecision(4).)

```
Enter the number to add the even numbers up to: 33
LOOP1 total = 272
Enter a lower and upper number to sum: 77 79
LOOP2 total = 234
Enter a file name to read from: in1.in
LOOP3 total = 18.0000
Do you want to do all of this again? y

Enter a file name to read from: in2.in
LOOP3 total = 14.0000
Do you want to do all of this again? y

Enter a file name to read from: in2.in
LOOP3 total = 14.0000
Do you want to do all of this again? n

Enter another file name to add 5 numbers from: in1.in
number3 = -1.0000
LOOP4 total = 18.0000

End of Program!
```

This shows what happens when the input file does not exist: (You should use a **break;** statement instead of using **"return 1;"** inside the **if** statement for when the file does not open so that the program continues on to the next step in the assignment.)

```
Enter the number to add the even numbers up to: 6
LOOP1 total = 12
Enter a lower and upper number to sum: 1 3
LOOP2 total = 6
Enter a file name to read from: in.in
```

```
error opening file!
```

```
Enter another file name to add 5 numbers from: in1.in
```

```
number3 = -1.0000
```

```
LOOP4 total = 18.0000
```

```
End of Program!
```

Submit Your Work

To submit your work to Web-CAT to be graded.