IEEE
Computer Chapter
Southern Alberta Section

SCHULICH
School of Engineering
UNIVERSITY OF CALGARY

IEEE Women in Engineering
WIE
Southern Alberta Section

SCHULICH IGNITE 2019

Session Overview

- Intro to **loops**
  - **while** loops
  - **for** loops

LÖÖPS

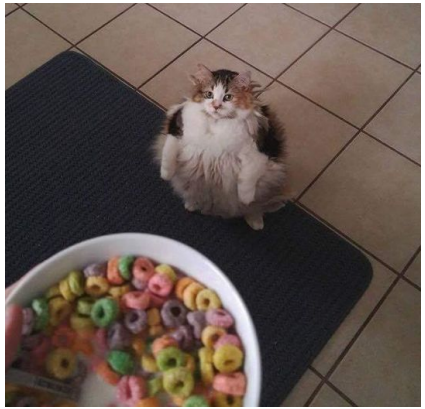

bröther may i have some lööps


Repeat
Repeat
Repeat
Repeat
Repeat
Repeat

WHILE LOOPS

- What if you want to run the same block of code a bunch of times, and then stop?

- To do this, we use **loops**



WHILE LOOPS

- A **while loop** works similarly to an **if** statement

```
while (condition == true){
    // Do something here
}
```

- As long as the *boolean expression* evaluates to true, the code inside will be repeated!

While vs If

`if` Statements

Runs **once**
if the condition is *true*

```
int money = 500;
if (money > 0) {

  println(“I buy it!”);
  money -= 100;
} else {

  println(“I am broke! :( ”);
}
```

`while` Loops

Runs **over and over**
while the condition is *true*

```
int money = 500;
while (money > 0) {

  println(“I buy it!”);
  money -= 100;
}

println(“I am broke :( ”);
```

CountDown

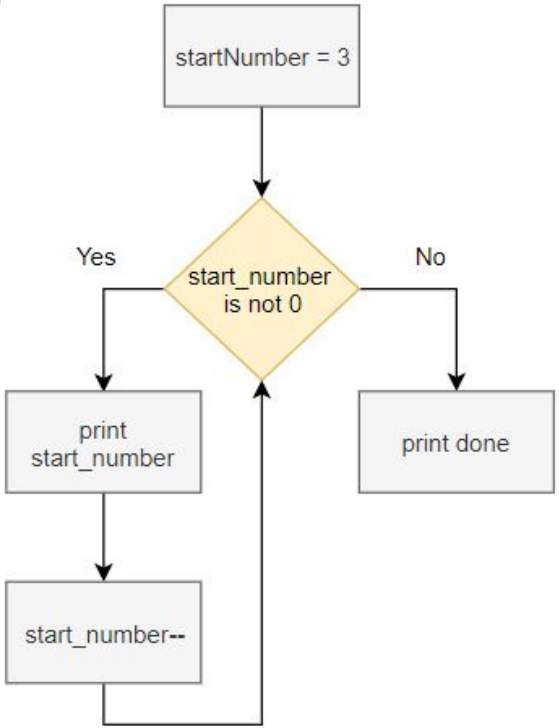
Consider a program which prints
all the numbers from 5 down to 0

- What condition should we check in our while loop?
- When do we want to stop looping?

Countdown Solution

The easiest way to write this is with a löop

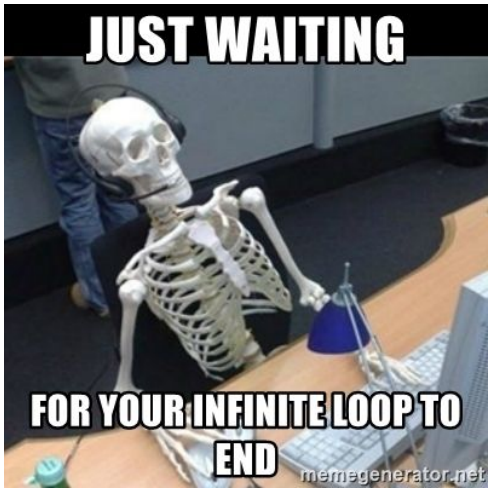
```
int startNumber = 5;
while(startNumber != 0) {
    println(startNumber);
    startNumber--;
}
println(“done”);
```



WHILE LOOPS

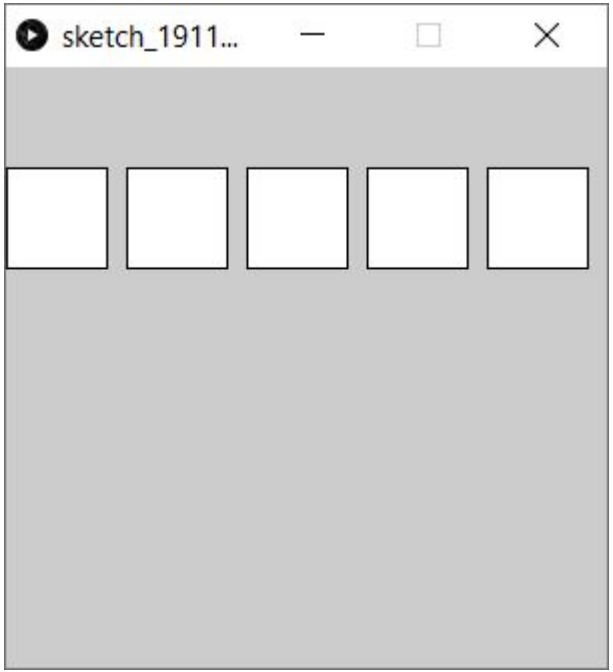
What happens if **startNumber** is -3?
Can anyone figure out what the problem is?

```
int startNumber = -3;
while(startNumber != 0) {
    println(startNumber);
    startNumber--;
}
```



Exercise 1: Squares

- Use a **while** loop to display 5 evenly spaced squares (Hint: the x position will depend on the loop counter)

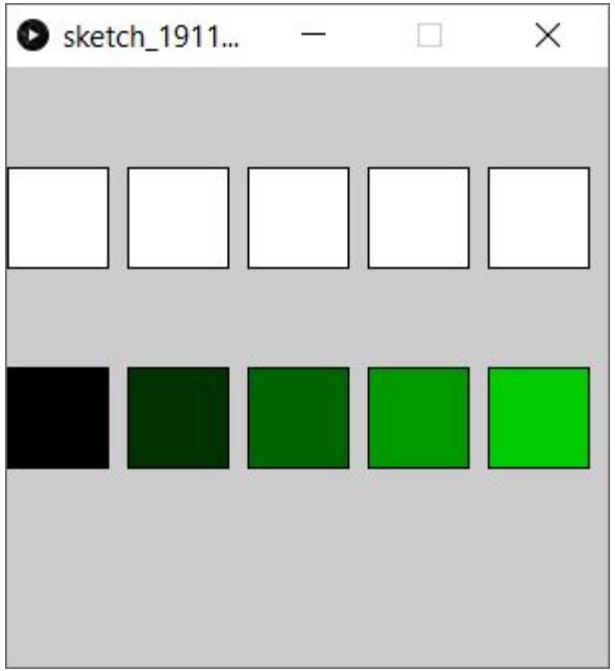


Exercise 1: Count Up

- Use a while loop to write a program that prints the numbers 0-5 to the console

Exercise 2: Fancy Squares

- Use a **while** loop to display 5 evenly spaced squares
  (Hint: the x position will depend on the loop counter)
- Add another **while** loop to make squares with colours!
  (Hint: the red/green/blue will depend on the loop counter)



For Loops

- A **for loop** is a *compact* way of writing a while loop with a *counter*

```
for(int i = 0; i < 10; i++) {
    println(i);
}
```

- The counter **i** cannot be accessed outside of the loop
  - It is **deleted** when the loop is done

For vs While

In any loop, there are usually three things we have to do:

CREATE; **CHECK;** **CHANGE**

```
int i = 0;
while(i < 10) {
    println(i);
    i++;
}
```

```
for(int i = 0; i < 10; i++) {
    println(i);
}
```

Putting it Together

A loop is often used with an array to get its values.

```
int[] myArray = {2, 4, 6, 8};

for (int i = 0; i < 4; i++) {
    println(myArray[i]);
}
```

Exercise 3

- Create an array with 5 elements,
  {1.2, 2.3, 3.4, 4.5, 5.6}
- Try **printing** your array with `println()`
  What happens?
- Find the *average*
  of the array values

Average Formula

$$\text{Average} = \frac{(a_1 + a_2 + ... + a_n)}{n}$$

Hint: Keep track of the sum of the elements!
You'll need a loop!

Exercise 4: Dartboard

- Draw a dartboard on the screen
- Use a loop to draw all of the circles at different radii
- Challenge: experiment with different colors!
  Hint: Try the random() function

Exercise 5: Lotsa Balls again

Take last week's example where you made 5 balls and add loops!

Recall:

- We made a class called **Ball**
- We made an array of **Ball** objects
- We used a *variable* as the **index**

Hint: You may want two loops
in **setup** (to create the balls)
and
in **draw** (to update the balls)