Introduction to Computer Science 1

Lecture 4 - Loops

d.camporaperez@maastrichtuniversity.nl

Maastricht University

Learning Goals

- You know how to use different kinds of loops
  - for
  - while
  - do while
- You understand the conditions under which loops run and stop
- You are familiar with the pitfalls of loops and how to avoid them
- You know how to write and recognize nested loops

Overview

Loops

- while
- do – while
- for

Loops

Something that repeats…

- Why?
  How would you print something like this?
  ```
  ****
  ****
  ****
  ****
  ****
  ****
  ```
- What if there were 1000 lines of ****?

Loops

Repetitive task

- computers are ideal for this
- don’t get tired or bored
- don’t make mistakes because of that
- they are fast at doing this

Real life loops

…

4. Pour the milk and cream into a pan and bring just to the boil. Remove from the heat. Add the chocolate and beat until it is melted and smooth with no lumps.
5. Gradually stir hot chocolate mix into paste. Return to pan. Cook, stirring, over a medium-low heat for 5 mins to a smooth, thick paste. Remove from the heat. Leave until cold, beating occasionally with a wire whisk.

…

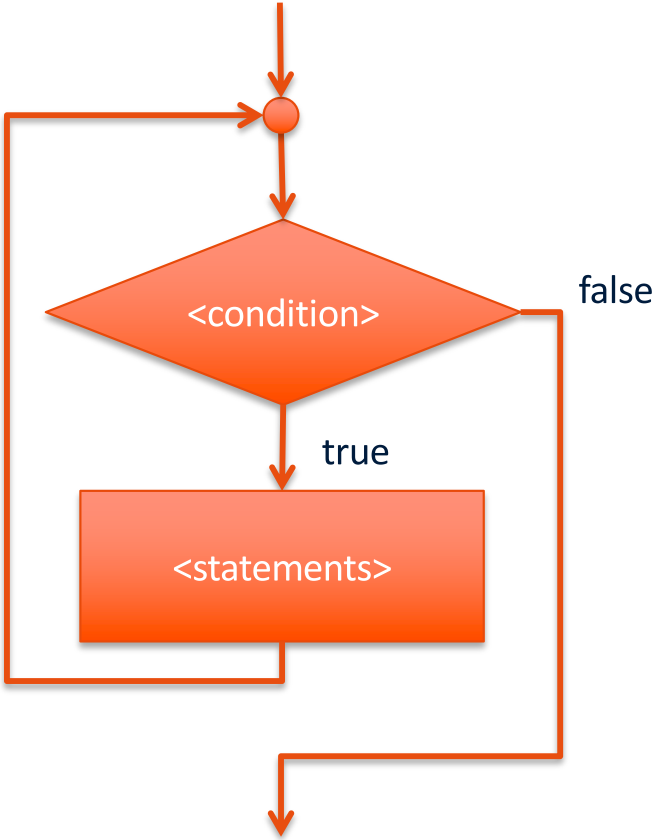
Graduate in 3 (or 5) years

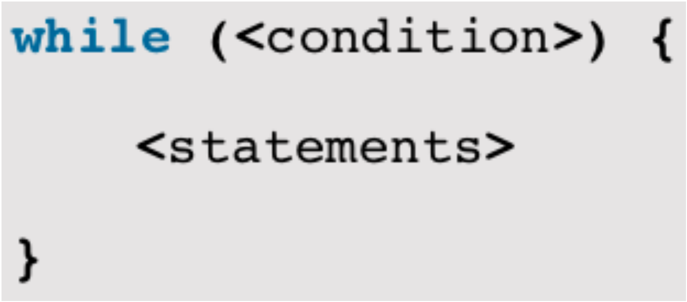
Walking (or traveling in general)

Swimming (running) 30 laps

…

WHILE





```
while (<condition>) {

      <statements>

}
```

You are my loop condition. I keep coming back to you

Exercise together: Counting kid

- Create a method of signature:

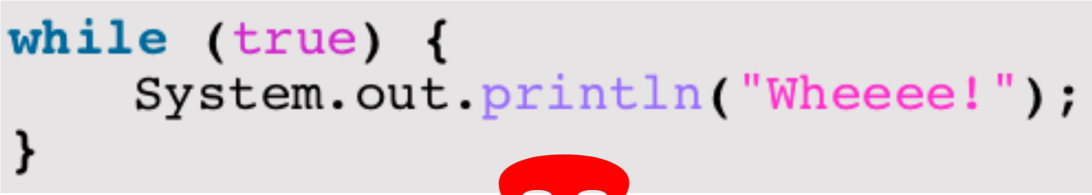
  ```
  public static void countingKid(int n)
  ```

- The method should print a character **n** times using a while loop.

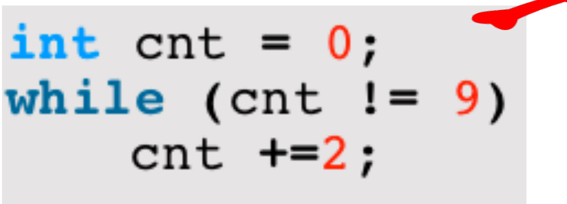
- When it is done, print “ Done!”

Frequent mistakes

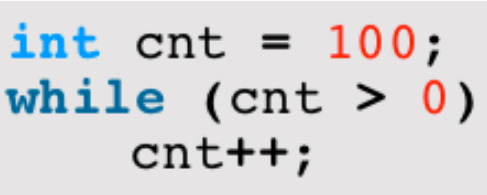
1. Infinite loops



```
while (true) {
     System.out.println("Wheeee!");
}
```



```
int cnt = 0;
while (cnt != 9)
     cnt +=2;
```



```
int cnt = 100;
while (cnt > 0)
      cnt++;
```

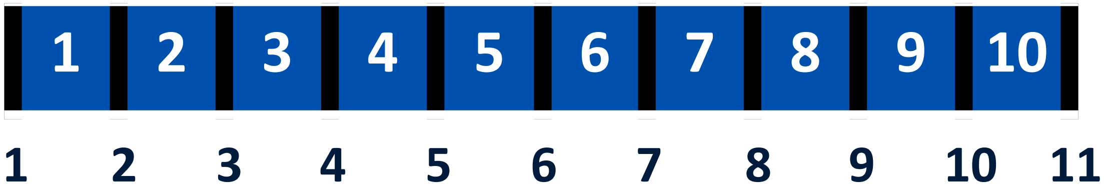
Frequent Mistakes (cont.)

2. Off-by-one

Boundary conditions

- initialization
- stopping condition

If you build a straight fence 100m long with posts 10m apart, how many posts do you need?



Exercise together: line reader

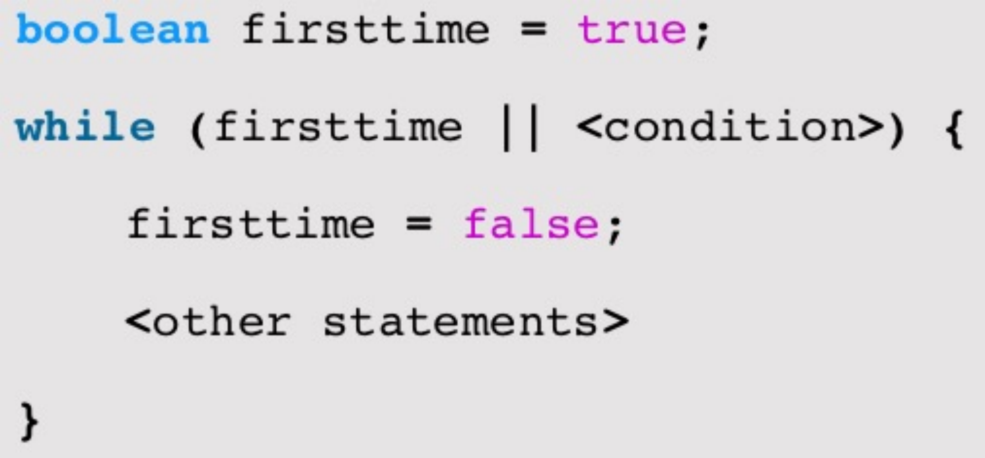
- Create a method of signature:
  `public static void readNumbers()`

- The method should greet the user with the message:
  `Enter a sequence of numbers, end with a letter:`

- It should then ask for numbers and print them until the user inputs something that is not a number

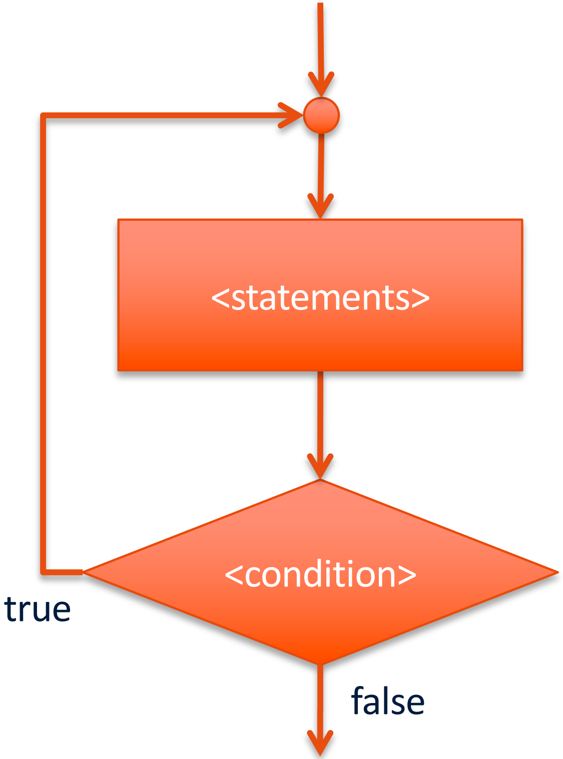
Running a loop at least once

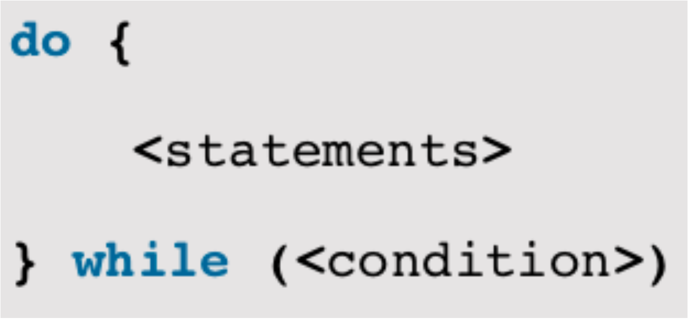
- While checks condition before starting loop
- What if you want to run the loop at least once?



```
boolean firsttime = true;

while (firsttime || <condition>) {

    firsttime = false;

    <other statements>

}
```

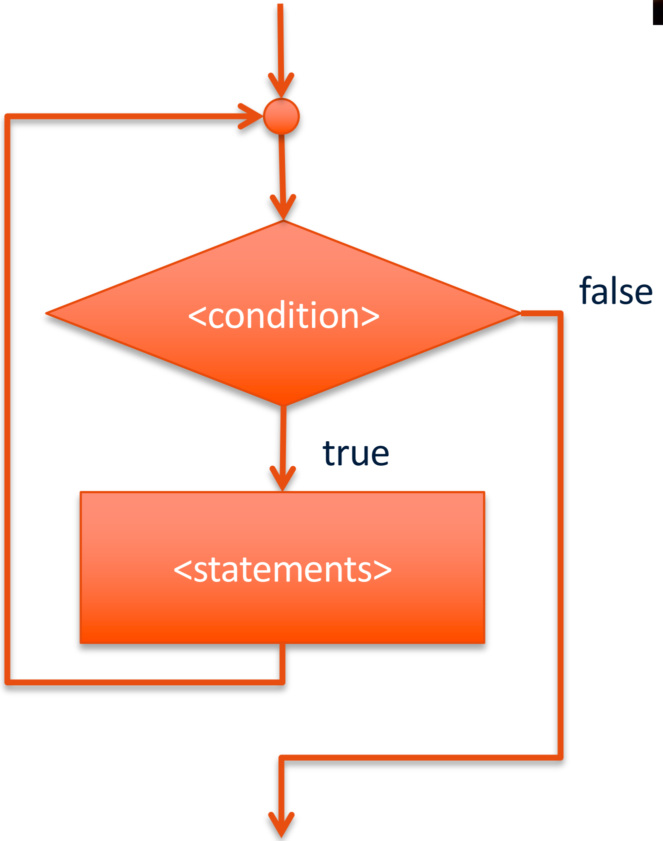
DO – WHILE

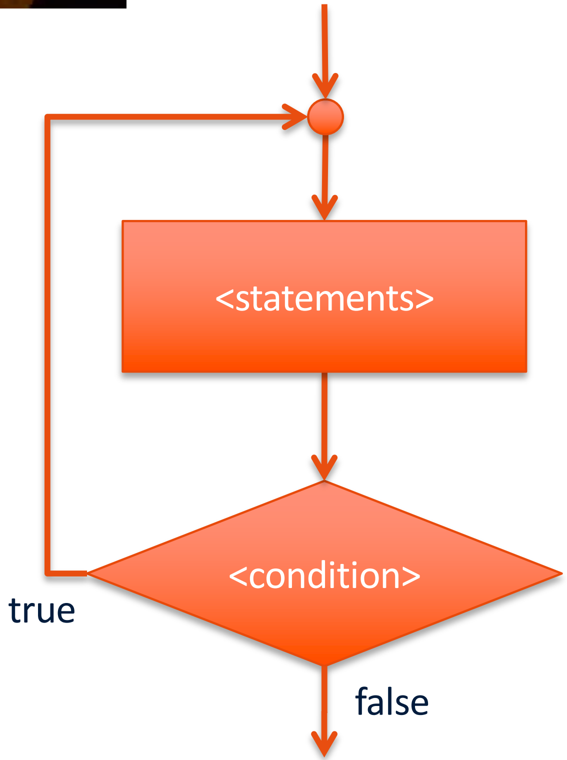




```
do {

    <statements>

} while (<condition>)
```

WHILE vs DO-WHILE





Exercise together: menu

- Create a method with signature
  - `public static int printMenu(Scanner input)`

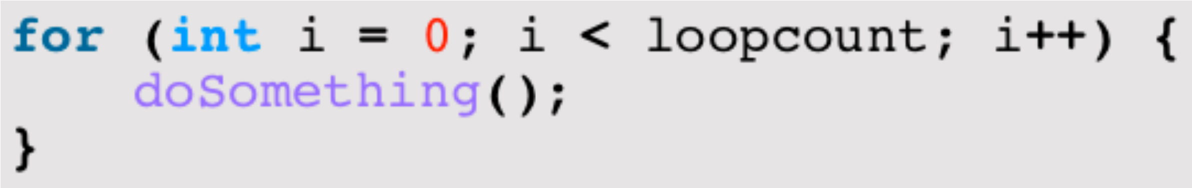
- The method should print:

  -
    ```
    Type 1 for option 1
    Type 2 for option 2
    Type 3 for option 3
    Type 0 to quit
    ```
  - It should get the user choice and return it as an int

- Use a do-while statement to continuously ask for the input of the user, until the user types in a 0 and the program quits

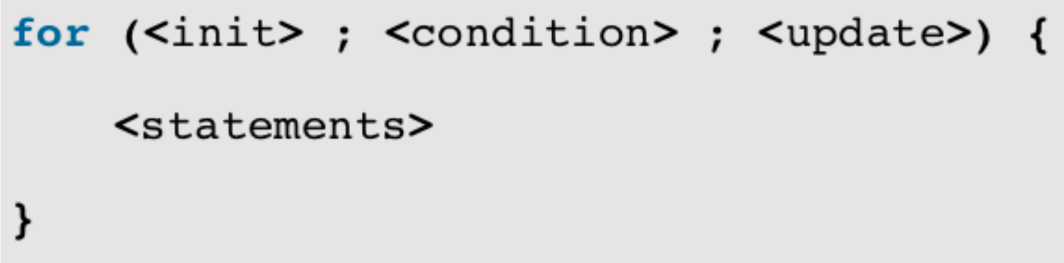
FOR

Syntactic sugar!



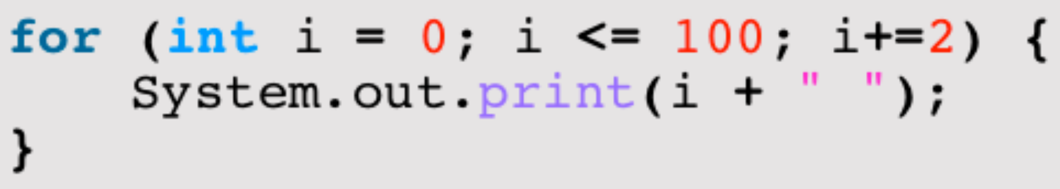
```
for (int i = 0; i < loopcount; i++) {
    doSomething();
}
```

In general:

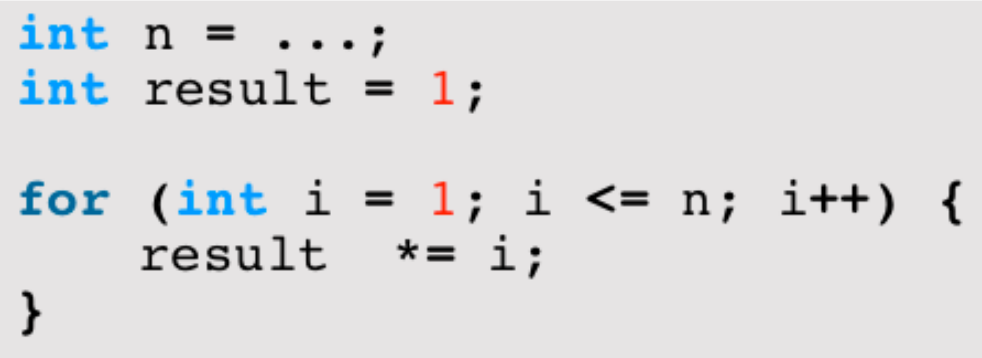


```
for (<init> ; <condition> ; <update>) {

    <statements>

}
```

For example



```java
for (int i = 0; i <= 100; i+=2) {
    System.out.print(i + " ");
}
```



```java
int n = ...;
int result = 1;

for (int i = 1; i <= n; i++) {
    result  *= i;
}
```

Exercise together: is prime

- Create a method of signature:

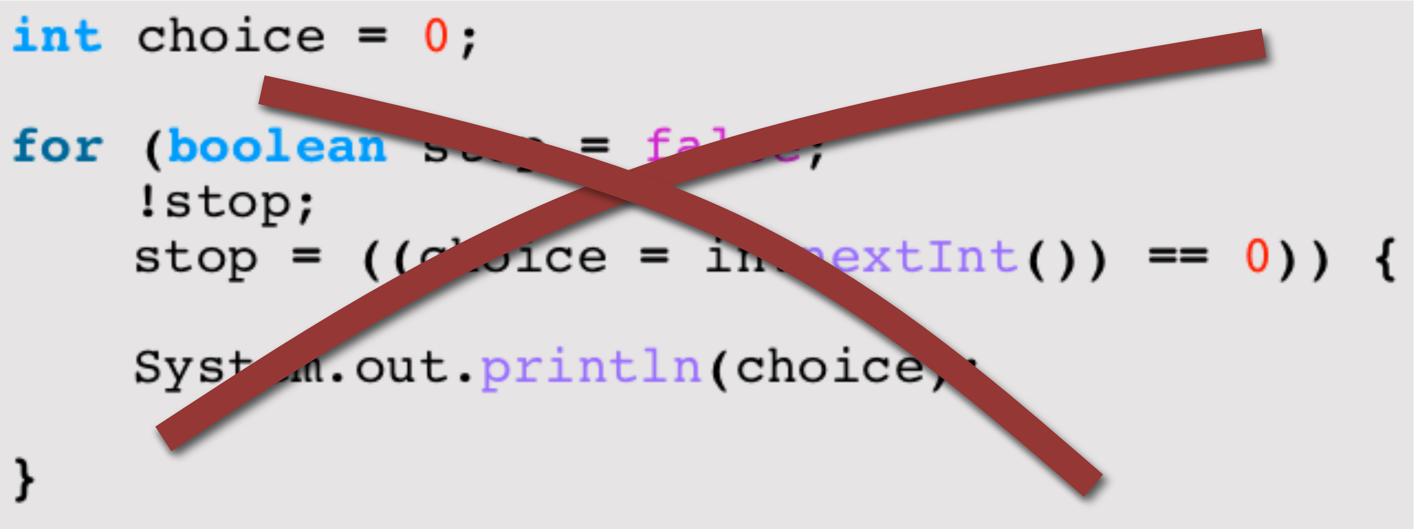
```
public static boolean isPrime(int n)
```

- The method should check whether integer **n** is prime

- It should return true if n is prime, and false otherwise

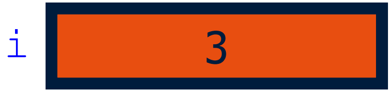
Bad form example

Use for-loops to emphasize structure!



```
int choice = 0;

for (boolean stop = false;
     !stop;
     stop = ((choice = in.nextInt()) == 0)) {

     System.out.println(choice);

}
```

Execution Trace: Counting loop



```
for ( int i = 0; i < 3; ++i) {
    System.out.println("i is " + i);
}

System.out.println("all done");
System.out.println("i is : " + i);
```

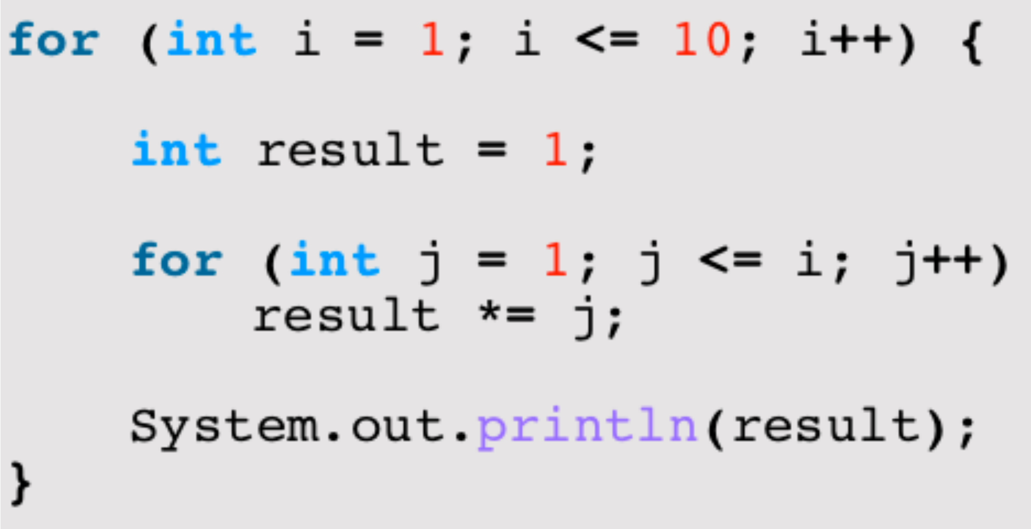
What is printed with this statement?

```
i is 0
i is 1
i is 2
all done
```

- Variables exist (live) only within the block they are defined
- In our example *i* has gone out of scope – it is *local* to the block (i.e. for loop) it is declared

Nested loops

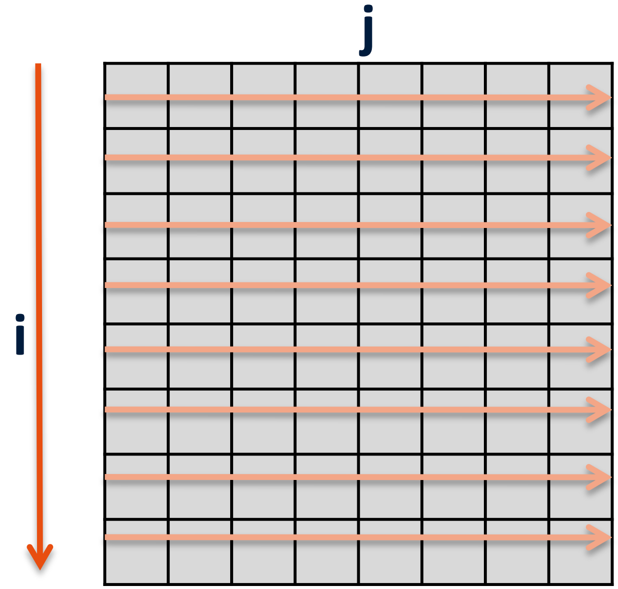
Just like if-statements, loops can be nested:



```
for (int i = 1; i <= 10; i++) {

    int result = 1;

    for (int j = 1; j <= i; j++)
        result *= j;

    System.out.println(result);
}
```

Nested Loops Examples

Matrices!



Matrices?

Exercise together: nested loops

- Create a nested loop that prints the following output:

-
```
0
 1
  2
   3
    4
     5
      6
       7
        8
         9
```

Summary

Loops

- While
- Do-while
- For

Books Chapters: Check Canvas

Quiz 4

Homework: ~17 tasks available

Coming up next

This week:

- Friday: Game Lab 2 & 3, Assignment 3

Next week:

- Arrays!