Lesson01 - Loops cont. (for loops)

This lesson is meant to be a follow-up to lesson00, (although retrospectively, it may be worthwhile to expand on while loops for another class period or two in order to solidify student's knowledge of it). As such, the same target audience is applicable here – students in grades 10-12 in an Intro to CS course or AP CS A.

Computer Science and Digital Fluency Standards: 9-12.CT.7, 9-12.CT.8, 9-12.CT.10

Overview

Structurally, this lesson is similar to lesson00. Students will start with a Do Now involving while loops as a warm-up for the rest of the lesson. The teacher will then transition to the slideshow with an introduction to for loops. Students will see a couple of examples of for loops, although in terms of the genereal concept of loops, they will have already established it during the previous lesson.

Next, students will do a quick trace diagrama activity. This is used to the same effect as the code tracing activity in lesson00 – it is a way for students to practice for loops without jumping right into writing their own programs. Students will once again "get into the mind" of a computer which will help solidify their learning. Again, this is also a good strategy for the teacher to see the pulse of the room.

After reviewing the trace diagrams, students should have a good idea of how for loops work, and have seen examples of for loops in action. This is where I decided to have them do a pair programming activity, involving a couple of for loops. The solution may not come to students very quickly because this is their first time working with for loops, which is why this is a collaborative activity. The code will be scaffolded with enough structure so that students can focus on the main point of the lesson. Students will have the rest of the class time to finish.

Intro (3-5 min)

Aim: What is a for loop and how can we use it in computer programming?

Do Now:

Review on while loops as a warm-up excercise. (Included in slides)

Short powerpoint slide/lecture to introduce loops (5 - 7 min)

Materials in folder

Trace diagrams of a for loop (7-10 min)

Directions/example in the slides. Class will come back together to show off the trace diagram (student volunteers and/or the teacher does it live).

Pair programming (25 min - end of class)

(w/ scaffolding)
This is under the assumption that students have already been introduced to pair programming. Students will work in pairs to complete two separate problems that require for loops. One student will be the navigator, and the other the driver. After completeing one of the two tasks, they will then swap roles. They have the rest of class to work on this.

For loops

Intro to CS

Aim: What is a for loop, and how can we use it in computer programming?

Do Now:

Mecha Mr. Zeng makes peanut butter & jelly sandwiches for Mr. Zeng during his free periods. Mr. Zeng wants to improve the mecha's peanut butter spreading efficiency, but first he needs to measure how long it takes for the mecha to spread the peanut butter.

In pseudo-code, write a while loop that will help keep track of the number of times mecha Mr. Zeng spreads peanut butter with it's knife, until the side of a piece of bread is fully covered in peanut butter.

```
int numberOfSpreads = 0;

  while (bread != fully spread) {
      spread peanut butter with knife;
      numberOfSpreads += 1;
  }
```

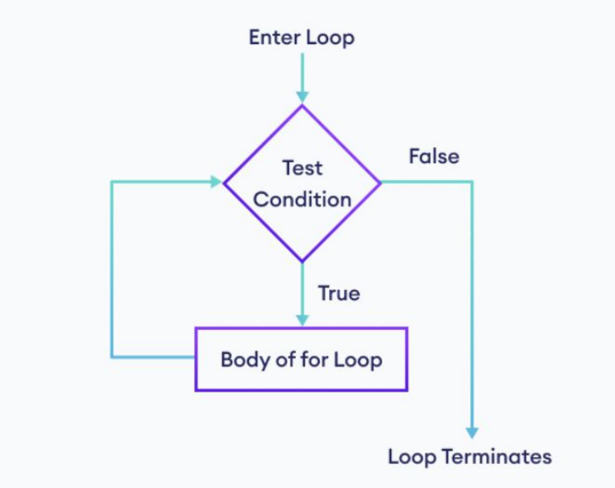
We use a while loop to make sure mecha Mr. Zeng finishes the job of spreading the peanut butter.

By creating an iterator “numberOfSpreads”, we can count every single time mecha Mr. Zeng performs the action “spread peanut butter with knife”.

Now Mr. Zeng can work on improving mecha Mr. Zeng!

For Loops

- For loops are a loop, just like while loops.

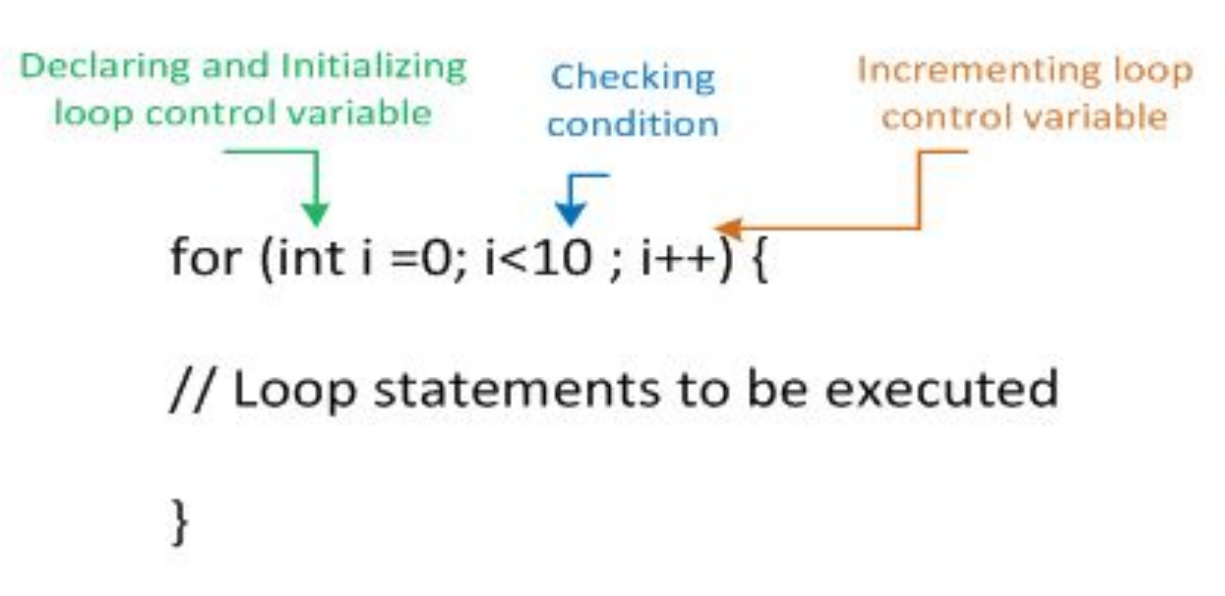


- Any loop that you write using a while loop can be written using a for loop, and vice versa. Functionally, they are the same: execute code as long as a condition is true.

- You can view for loops as a more concise way to write while loops with iterators. The three components of a for loop are the initializer of an iterator, the condition and the increment/decrement of the iterator.

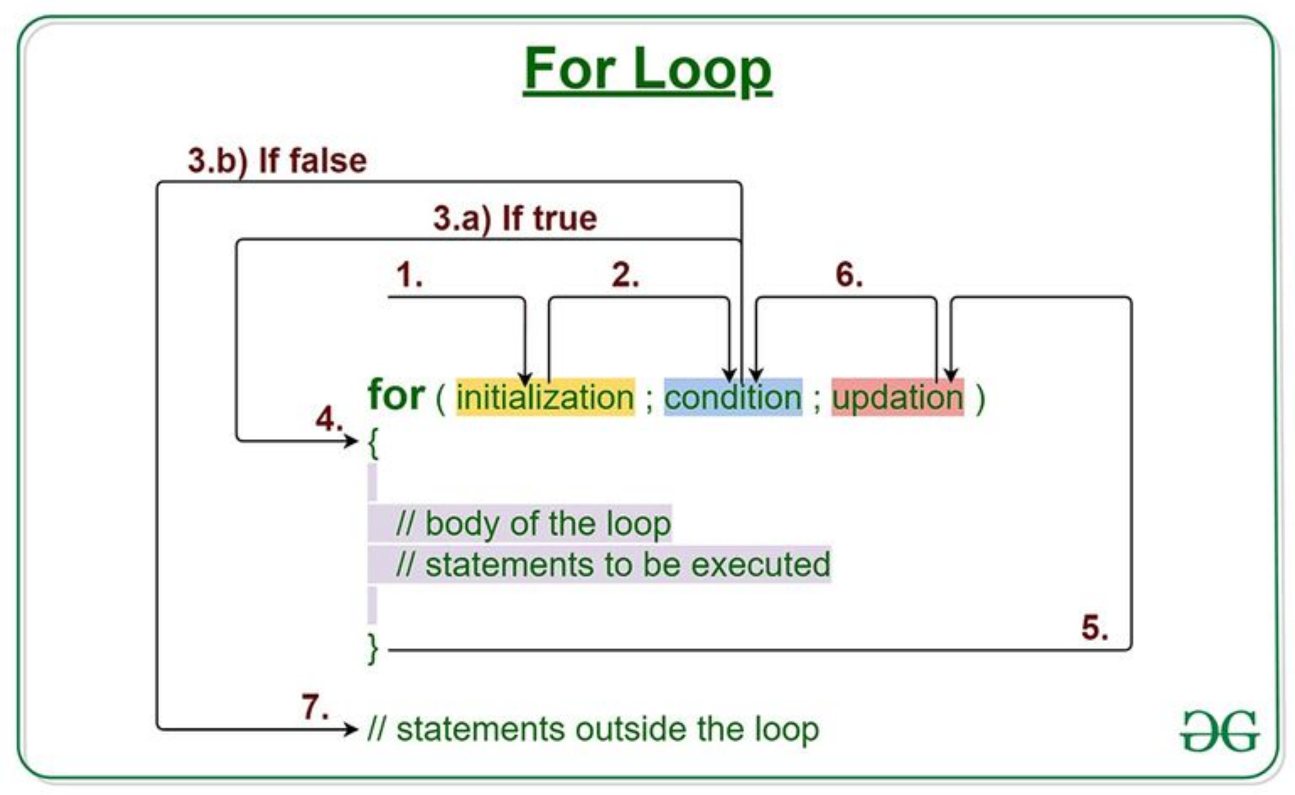
For Loops - cont.

Note: in this case, the loop control variable, "i", stays INSIDE the loop!


Declaring and Initializing loop control variable
Checking condition
Incrementing loop control variable

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

// Loop statements to be executed

}
```

- As you can see, for loops are "cleaner" as put all the function of iteration in a single line of code.
- For loops are very, very commonly used to traverse through arrays in computer programming, but you'll learn more about what an array is later.



For Loop

```
for ( initialization ; condition ; updation )
{

   // body of the loop
   // statements to be executed

}
// statements outside the loop
```

Let’s look at the for loop version of a while loop from last class…

For loop template:

for (the initializer; the condition ; the update)

```
int i = 1;
while (i <= 5) {
    System.out.println(i);
    i++;
}
```

This program prints out:

```
1
2
3
4
5
```

We know this program runs a line of code exactly 5 times, and we also already know what it’s going to output. A lot of the work is already done since we have the while loop, which even had an iterator!

Take a minute to write out the for loop version of this program.

Answer: for (i

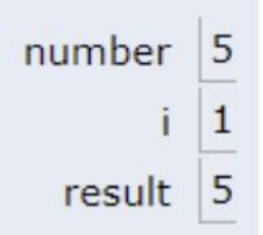
drag me!

Trace Diagrams

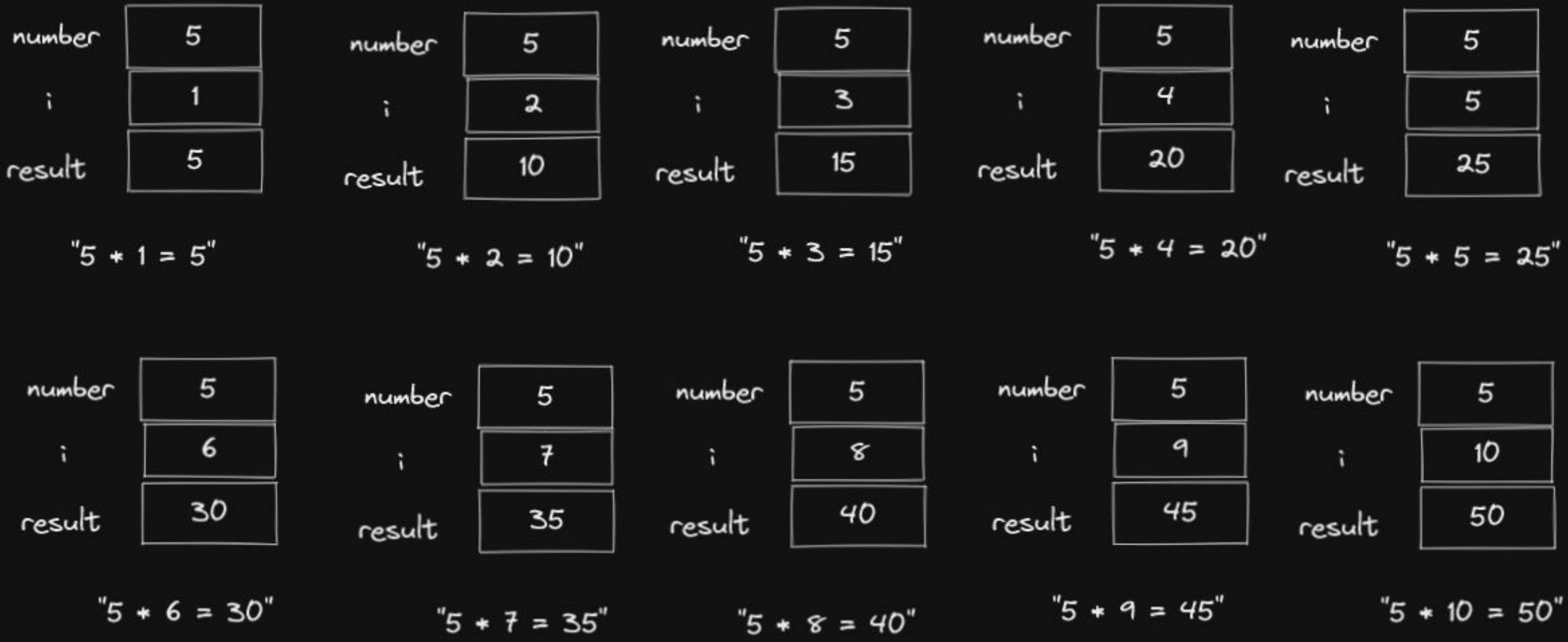
```
int number = 5;

 for (int i = 1; i <= 10; i++) {
     int result = number * i;
     System.out.println(number + " x " + i + " = " + result);
 }
```

Trace through this program on a piece of paper. Make sure to keep track of where the computer is during each step of the program, as well as keep track of what the value of i and result are during each step.

Similar to this!

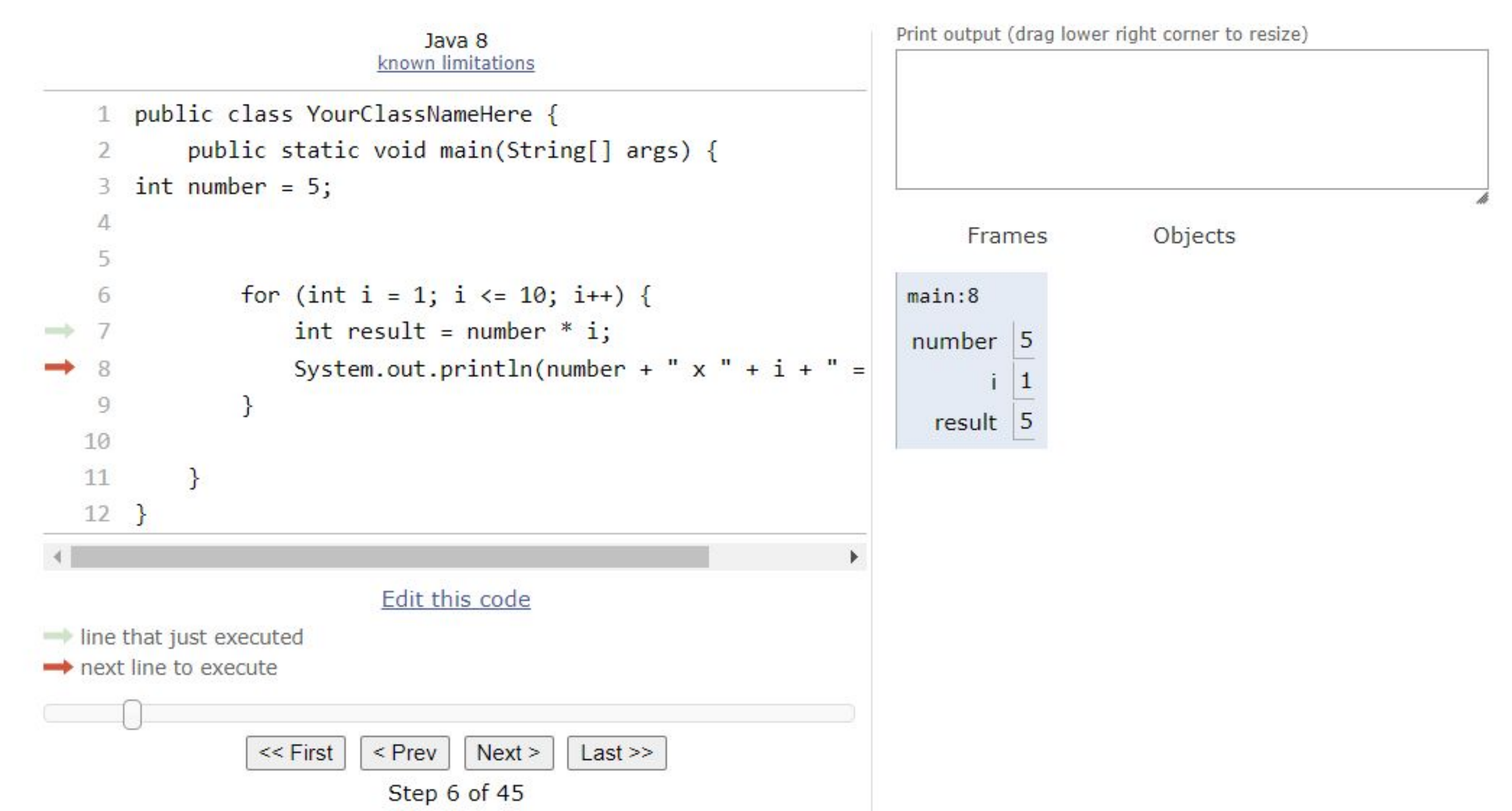


number	5
i	1
result	5



```
"5 * 1 = 5"
"5 * 2 = 10"
"5 * 3 = 15"
"5 * 4 = 20"
"5 * 5 = 25"
"5 * 6 = 30"
"5 * 7 = 35"
"5 * 8 = 40"
"5 * 9 = 45"
"5 * 10 = 50"
```

A cool resource that shows a visualization of code online!



Java 8

known limitations

```
public class YourClassNameHere {
    public static void main(String[] args) {
int number = 5;


        for (int i = 1; i <= 10; i++) {
            int result = number * i;
            System.out.println(number + " x " + i + " =
        }

    }
}
```

Edit this code

➡ line that just executed

➡ next line to execute

Step 6 of 45

Print output (drag lower right corner to resize)

main:8

number	5
i	1
result	5

https://pythontutor.com/java.html#mode=edit

Pair Programming

Work with the person sitting next to you!
One person will be the navigator and the other will be the driver.

When you finish writing the code for the first part, switch!
You can comment out question 1 and then uncomment the starter code for question 2.

Student version

Students are provided a skeleton code/scaffolded code. If you suspect your students would want a challenge, or know that certain students would want more of a challenge, remove all skeleton code/scaffolding.

```
import java.io.*;

public class Main {

// write some code in main using a for loop that will calculate
        the sum of all numbers from 1 to a number "n". Test it
        by finding the sum of all numbers from 1 to 100.
       public static void main(String[] args) {
                int n;
                int sum;                 //use this variable to
        store the value of the factorial as the program iterates
        through the for loop

                for (                     ) {

                }

        System.out.println(                   );
            }

/*
// write some code in main using a for loop that will calculate
        factorial of the number n. Test it by finding the
        factorial of 10.
        public static void main(String[] args) {
                int n;
                int factorial;           //use this variable to
        store the value of the factorial as the program iterates
        through the for loop

                for (               ) {

                }
                System.out.println(                );
            }

*/

/*
Expected output for sum:
Sum of numbers from 1 to 100 is: 5050
```

```
Expected output for factorial:
Factorial of 10 is: 3628800

*/
```

Teacher version/ answer key

```
import java.io.*;

public class Main {

       public static void main(String[] args) {
                int n = 100;
                int sum = 0;

                for (int i = 1; i <= n; i++) {
                    sum += i;
                }

        System.out.println("Sum of numbers from 1 to " + n + "
        is: " + sum);
            }

/*


        public static void main(String[] args) {
                int n = 10;
                int factorial = 1;

                for (int i = 1; i <= n; i++) {
                    factorial *= i;
                }
                System.out.println("Factorial of " + n + " is: "
        + factorial);
            }

*/
}
```