Loops and Arrays

In this course, you will often work in teams of 3–4 students to learn new concepts. The "meta" activity will introduce you to the team framework. Then, we'll look at how to structure ***loops*** in Java. Finally, programs often need to store multiple values of the same type, such as a list of phone numbers or the names of students. Rather than create a separate variable for each one, we can store them together using an ***array***.

Manager: Recorder:

Presenter: Reflector:

Content Learning Targets

After completing this activity, you should be able to say:

- I can describe the responsibility of each role and summarize the benefits of teamwork.
- I can identify the three main components of a while loop.
- I can rewrite a while loop as a for loop (and vice versa).
- I can declare and initialize array variables of primitive types.
- I can iterate (i.e., loop) over arrays of primitives.

Process Skill Goals

During the activity, you should make progress toward:

- Leveraging prior knowledge and experience of other students. (Teamwork)
- Justifying answers based on evidence provided in the model. (Problem Solving)
- Tracing execution of while/for loops and predicting their output. (Critical Thinking)

Meta Activity: Team Roles

Decide who will be what role for today; we will rotate the roles each week. If you have only three people, one should have two roles. If you have five people, two may share the same role.

Manager:
Presenter:
Recorder:
Reflector:

Questions (15 min) Start time:

1. What is the difference between **bold** and *italics* on the role cards?

2. Manager: invite each person to explain their role to the team. Recorder: take notes of the discussion by writing down key phrases next to the table above.

3. What responsibilities do two or more roles have in common?

4. For each role, give an example of how someone observing your team would know that a person is <u>not</u> doing their job well.

- Manager:
- Presenter:
- Recorder:
- Reflector:

Model 1 Assignment

Consider the following Java statements. What is the resulting value of each variable?

A:

```
int x, y;
x = 1;
y = 2;
y = x;
x = y;
```

Value of x:

Value of y:

B:

```
int x, y, z;
x = 1;
y = 2;
z = y;
y = x;
x = z;
```

Value of x:

Value of y:

Value of z:

C:

```
int z, y;
z = 2;
z = z + 1;
z = z + 1;
y = y + 1;
```

Value of z:

Value of y:

Questions (15 min) Start time:

5. In program A, why is the value of `x` not 2?

6. In program B, what happens to the values of `x` and `y`?

7. In program B, what is the purpose of the variable `z`?

8. If program C runs, what happens to the value of `z`?

9. In program C, why is it possible to increment `z` but not `y`?

10. Because ***increment*** and ***decrement*** are so common in algorithms, Java provides the operators `++` and `--`. For example, `x++` is the same as `x = x + 1`, and `y--` is the same as `y = y - 1`. Write the value of `x` and `y` next to each statement below.

```
int x = 5;
x--;
x--;
```

```
int y = -10;
y++;
y++;
```

11. Like the assignment operator, the `++` and `--` operators replace the value of a variable. Java also has ***compound assignment*** operators for convenience. For example, the statement `x = x + 2` can be rewritten as `x += 2`. Simplify the following assignment statements.

```
step = step + 5;
size = size - 3;
total = total * 2;
change = change / 10;
hours = hours % 24;
```

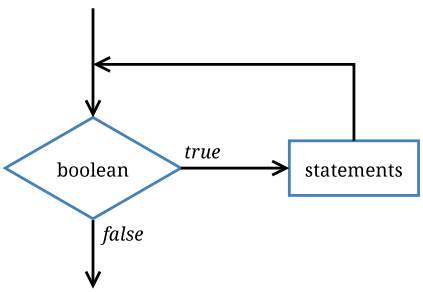
12. Which of the following assignment statements can also be rewritten like the ones in #11?

```
step = 5 + step;
size = 3 - size;
total = 2 * total;
change = 10 / change;
hours = 24 % hours;
```

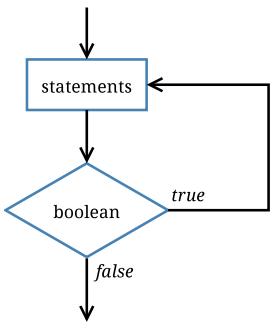
Model 2 While Loops

A loop is a set of instructions that are to be repeated. All loops have three main components: ***initialize***, ***test***, and ***update***. Identify these components in each of the examples below.

```
// pre-test loop
number = 1;
while (number <= 10) {
    System.out.println(number);
    number++;
}
```




```
// post-test loop
number = 1;
do {
    System.out.println(number);
    number++;
} while (number <= 10);
```



Questions (20 min) Start time:

13. Which loop component always happens first? Why?

14. Explain why the `while` loop is called a ***pre-test*** and the `do while` loop is called a ***post-test***.

15. What is output to the screen by each loop? Predict the output first, and then run the code to check your answer.

16. What is the final value of `number` at the end of each loop? Make a prediction first, and then add print statements to the code to check your answer.

17. How does the output change if you swap the `println` and `number++` statements?

18. What is the output to the screen if you remove the `number++` statement?

19. What is the difference between a `while` statement and an `if` statement?

20. What is output by the following loop? Explain how the code works.

```
number = 99;
do {
    System.out.println(number);
    number++;
} while (number <= 10);
System.out.println(number);
```

21. What is output by the following loop? And what mistake was made?

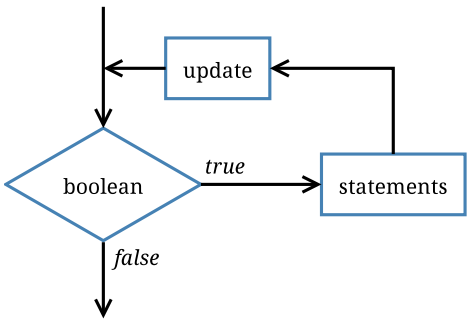
```
int i = 0;
while (i < 3)
    System.out.println("i = " + i);
    i = i + 1;
```

Model 3 For Loops

The `for` loop combines ***initialize***, ***test***, and ***update*** into one line of code.

```
// Loop A: count forwards
for (i = 1; i <= 10; i++) {
    System.out.println(i);
}

// Loop B: count backwards
for (i = 10; i >= 1; i--) {
    System.out.println(i);
}
```



Questions (10 min) Start time:

22. Identify the components of each `for` loop.

Loop A:

a) initialize

b) test

c) update

Loop B:

a) initialize

b) test

c) update

23. Rewrite each `for` loop as a `while` loop.

Loop A: **Loop B:**

24. What do each of the `for` loops output to the screen? Be specific.

25. Describe how to change the `for` loops to print even numbers only (i.e., the output should be 2 4 6 8 10 and 10 8 6 4 2).

26. In mathematics, the factorial of an integer n (denoted by $n!$) is the product of all positive integers less than or equal to n. For example, the factorial of 5 is:

$$5! = 5 * 4 * 3 * 2 * 1 = 120$$

The following code computes the factorial of 5:

```
fact = 1;
i = 5;
while (i > 1) {
    fact *= i;
    i--
}
```

a) Rewrite the code above using a `for` loop instead of a `while` loop.

b) How would you change the code to compute the factorial of 12?

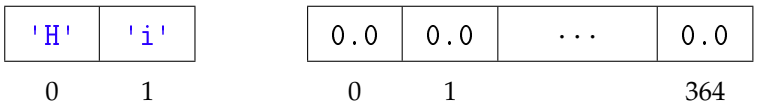
Model 4 Array Syntax

An ***array*** variable allows you to store multiple variables (of the same type). Each value in an array is known as an ***element***. The programmer must specify the ***length*** of the array (the number of array elements). Once the array is created, its length cannot be changed.

```
char[] letterArray = {'H', 'i'};
System.out.println(letterArray[0]);          // outputs H
System.out.println(letterArray.length);      // outputs 2

double[] numberArray = new double[365];
System.out.println(numberArray[0]);          // outputs 0.0
System.out.println(numberArray.length);      // outputs 365
```

Array elements are accessed by ***index*** number, starting at zero:



Questions (5 min) Start time:

27. Examine the results of the code.

a) What is the length of `letterArray`?

b) What is the length of `numberArray`?

c) What is the index of the element `'i'` in `letterArray`?

d) What is the index of the last element of `numberArray`?

28. Now examine the syntax of the code.

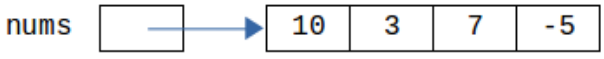
a) What are three ways that square brackets [] are used?

b) In contrast, how are curly braces {} used for an array?

Model 5 Array Diagrams

Array elements are stored together in one contiguous block of memory. To show arrays in box-and-pointer memory diagrams, we simply draw adjacent boxes.

```
int[] nums = {10, 3, 7, -5};
```




```
double[] stats = new double[3];
```

Questions (15 min) Start time:

29. What is the default value for array elements?

30. Draw a memory diagram for the following arrays. (*Hint:* You should have no empty boxes.)

a)
```
int[] sizes = new int[5];
sizes[2] = 7;
```

b)
```
double[] costs = new double[4];
costs[0] = 0.99;
```

31. Arrays are ***reference types***, not primitives. What is the ***value*** of an array variable?

32. Does the statement `int[] copy = nums;` create a new array? Justify your answer.

33. Draw a memory diagram of the following array.

```
double[] prices = new double[4];
prices[1] = 10 / 4;
prices[3] = 3.49;
```

Model 6 Array Iteration

We can use loops with arrays to fill in initial values or access/modify current values. First, work these problems out on a whiteboard or paper. Once your team has reached a consensus, check your answers using `ArrayIteration.java`.

Questions (15 min) Start time:

34. Write a Java for loop to iterate through an array of integers named `arr` and add their values, storing the total in an integer called `sum`. (***Hint:*** How do you know when to stop?)

35. Write a Java for loop to populate an empty (i.e., zeroed-out) array of integers named `triangles` with the triangular numbers up to the length of the array, starting with 0.

36. Write a java while loop to find the first negative value in an array of doubles called `earnings`. Print both the index and the value. if all values are nonnegative, print the message `No negative earnings.`