

BIT115: Introduction  
to Programming

**Lecture 20**

**Part 1**

Instructor: Craig Duckett



*"The Last Lecture"*

# Assignment Dates (By Due Date)

- **Assignment 1 (LECTURE 5)**  
Section 1: Wednesday, October 11<sup>th</sup>  
Section 3: Thursday, October 12<sup>th</sup>
- **Assignment 2 (LECTURE 9)**  
Section 1: Wednesday, October 25<sup>th</sup>  
Section 3: Thursday, October 26<sup>th</sup>
- **Assignment 1 Revision (LECTURE 11)**  
Section 1: Monday, November 6<sup>th</sup>  
Section 3: Thursday, November 2<sup>nd</sup>
- **Assignment 2 Revision (LECTURE 13)**  
Section 1: Monday, November 13<sup>th</sup>  
Section 3: Thursday, November 9<sup>h</sup>
- **Assignment 3 (LECTURE 15)**  
Section 1: Monday, November 20<sup>th</sup>  
Section 3: Thursday, November 16<sup>th</sup>
- **Assignment 3 Revision (LECTURE 18)**  
Section 1: Wednesday, November 29<sup>th</sup>  
Section 3: Thursday, November 30<sup>th</sup>

- **Assignment 4 (LECTURE 21) NO REVISION AVAILABLE!**  
Section 1: Monday, December 11<sup>th</sup>  
Section 3: Tuesday, December 12<sup>th</sup>



The Fickle  
Finger of Fate

# Course/Instructor Evaluations

BIT115

<http://www.cascadia.edu>

The screenshot shows the Cascadia Community College website. At the top, the navigation menu includes 'Registration', 'Schedule', 'Key Dates', 'Online Classrooms', 'Canvas', 'Student Toolbox' (circled in red), and 'Search Cascadia'. Below the navigation is a banner with 'CASCADIA COLLEGE BOTHELL' and links for 'DISCOVER CASCADIA', 'INSTRUCTIONAL PROGRAMS', 'FUTURE STUDENTS', and 'CURRENT STUDENTS'. The main content area is titled 'STUDENT TOOLBOX' and contains a list of links: 'Account Management', 'Address Change', 'Class Schedule and Catalog', 'CORE (Cascadia's Orientation and Registration Experience)', 'Course Evaluation' (circled in red), and 'Degree Audit'. A red circle highlights the 'Course Evaluation' link, and another red circle highlights the text 'Click here for Course Evaluation.' below the list.

The screenshot shows the Cascadia Community College login page. It features the college's logo and the text 'The resource that you have attempted to access requires that you be logged in.' Below this, there are input fields for 'Username:' and 'Password:', followed by a 'Login' button. A 'Logout' button is also visible. The page includes a 'LOG OFF' instruction: 'Exit your Web browser to end your connection.' and contact information for the Open Learning Center. At the bottom, there is a copyright notice: 'Copyright © 2009 Cascadia Community College • 1834'. The background of the page features faint, stylized text including 'Engineering', '58-acre wetlands', 'STUDENT-CENTERED', 'COMMUNICATE', 'ENVIRON', 'PRO', 'ESL', and 'COL'.

# Today's Topics

- *Passing and Returning Arrays*
- *The String Class*
- *Extra Credit*

# But first ...

## The Last Quiz



BIP 115 Lecture 13 Quiz Name: \_\_\_\_\_  
Wednesday, August 15, 2012

Add the missing Java source code to the following file so that everything in the array is printed to the screen.

```
import java.util.*;

class ArrayHelper {
    public void PrintArray(_____) {
        for(_____) {
            System.out.println(____);
        }
    }
}

public class Test extends Object {
    public static void main(String[] args) {
        int[] numbers = new int[3];
        numbers[0] = -10;
        numbers[2-1] = 42;
        numbers[5/3] = 13;

        ArrayHelper ah = new ArrayHelper();
        ah.PrintArray(numbers);
    }
}
```

**Output**

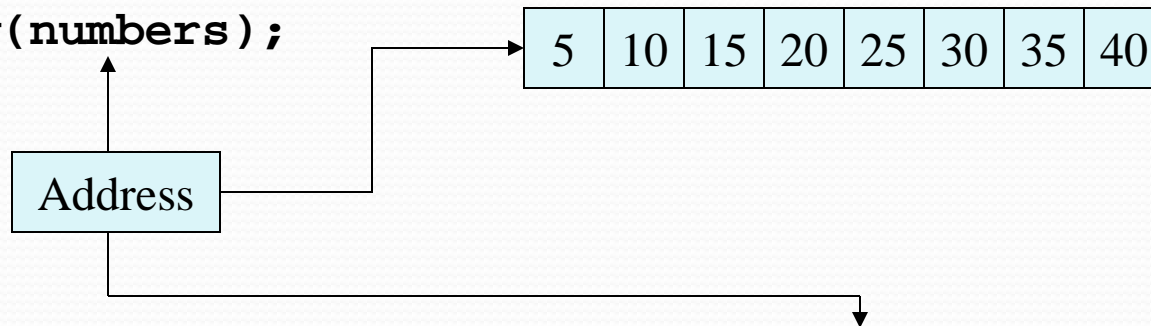
A screenshot of an IDE's output window. The window title is "ArrayHelper ah = new ArrayHelper();". The output text shows "JRE1.6.0\_20: operation complete." and "JRE1.6.0\_20: operation complete." The IDE interface includes a toolbar with icons for Run, Stop, and other actions, and a "Compile Messages" section on the left.

# Passing and Returning Arrays

## Passing Arrays as Arguments

- Arrays are objects.
  - Their references can be passed to methods like any other object reference variable.
- 

`showArray(numbers);`



```
public static void showArray(int[] array)
{
    for (int i = 0; i < array.length; i++)
        System.out.print(array[i] + " ");
}
```



# Passing and Returning Arrays

In **Last Wednesday's Class** we learned how you could **pass** an array into a method. Today we're going to take this to the next step to learn how you can **return** an array from a method.

A method can return a reference to an array. To do so, the return type of the method must be declared properly. For example, look at the following method definition:

```
public static double[] getArray() // <-- No parameters called  
{  
    double[] array = {1.2, 2.3, 4.5, 6.7, 8.9}  
    return array;  
}
```

**double** is the return type

**return** sends back a double

- The **getArray** method is a public static method that returns an array of doubles.

See example: [ReturnArray.java](#)

# ICE: Arrays as Return Values

## bit115> Lecture 20

In-Class exercises: Arrays as Return Values