Use the following outline to guide your self-assessment and notetaking

Week 6 – Arrays

Array Elements (Ch 8.1)

Difference from ArrayList

- Arrays have a fixed size that cannot be changed once set, whereas arraylists are variable
- Arrays are more efficient because they grab data at indexes rather than at references

Declaring and Using Arrays (Ch 8.2)

Object status

- All objects must be instantiated (including the array itself)
- Arrays can only hold one type (cannot have an array that holds both ints and Strings)
- Arrays can hold either primitive or object (class) types
- Size must be predetermined and specified

Bounds Checking

- Ensures that the index is in ranged for the array being referenced
- Must be in between 0 (inclusive) and the size of the array (exclusive)
    - Otherwise, an ArrayOutOfBounds exception is thrown
- Length constant is an attribute of the array and is used to check the size of the array

Alternate Array Syntax

- 2 ways to declare an array:
    - Ex: int [] grades and int grades []
    - No difference between these but first one is more common and consistent with other different types of declarations

Initializer lists

- Can be used to instantiate an array and provide initial values for the elements of the array
- Separated by commas and delimited by braces
- Size is determined by the number of items in the initializer list

Arrays as parameters

- An array can be passed as a parameter to a method
- Copy of the reference to the original array is passed
- Method can change an element of the array, but it cannot permanently change the reference to the array itself
- An element of an array can also be passed in

    - If the element is a primitive, a copy of the value is passed
    - If the element is a reference to an object, a copy of the object reference is passed

Arrays as Objects (Ch 8.3)

Memory implications

- Design of a program should be able to combine object representations that contain variables (some could be arrays) and methods
- Each element is a separate object (arrays of objects are referring to each object's reference)
    - An array of strings does not create any string objects
    - Only holds the references to string objects

Command Line Arguments (Ch 8.4)

Useful for…

- Providing input to a program without requesting explicitly from the user
    - If less input is provided on the command line, the program will throw an ArrayIndexOutOfBoundsException. But if more input is provided, it will be stored in the args array but ignored by the program

Variable Length Parameter Lists (Ch 8.5)

Useful for

- Inputting as many parameters as we want through methods
- Parameters are put into an array for processing within the method
- But parameters have to be the same type
- Must come last in the formal arguments and cannot accept 2 sets of varying parameters

Two-dimensional arrays (Ch 8.6)

Useful for…

- A more matrix-like representation of values
- Basically, an array of arrays
- A nested loop is needed to iterate over all the values
    - Outer loop is for each array in the 2d array (rows)
    - Inner loop is for the values in each row (columns)
- Initializer lists can be used to instantiate a 2d array (just need more sets of curly braces)

Multi-dimensional arrays

- An array can have as many dimensions as you want

- Each subsequent dimension is a subdivision of the previous one
    - Ex: number of students attending universities across the U.S
        - 1st dimension may be state
        - 2nd may be universities in each state
        - 3rd may be colleges in each university
        - 4th may be the departments in each college
            - This is where the number of students may be stored
- Using an array like this gets very complex and in OOP, anything higher than 2 dimensions is rare