

Array Review

Recall from ICS3U:

- An *array* is a structure that can store many of the same kind of data together at once.
- Arrays are an important and useful programming concept because they allow a collection of related values to be stored together with a single descriptive name
- Arrays have a **fixed length**
- Each data item in an array is called an *element*
- Each element has an *index* value (first is 0, then 1 ..)
- Declared using []

3 ways to declare an array of Strings:

declares an array of strings called friends to hold 5 strings

```
String[] friends = new String[5]
```

OR

declares an array of strings called friends to hold as many strings as the user wants

```
numfriends = input.nextInt();  
String[] friends = new String[numfriends];
```

OR

declares an array of strings called friends to hold these 5 strings

```
String[] friends = {"Kermit", "Lucille", "Sammy", "Roxy", "Myah"};
```

Using Arrays

- Access an array element by including its index in brackets after the array name.

```
System.out.println(friends[2]);    displays Sammy  
friends[2] = "Sunshine";          changes Sammy to Sunshine  
friends[5] = "McDougall";          ERROR — past bounds of the array  
numElements = friends.length();    numElements will be 5  
  
for (int i = 0; i < friends.length(); i++) { loops through array to display all names  
    System.out.println(friends[i]);  
}
```

Array Parameters

- A method declaration can include array parameters
- An entire array can be passed or just an element of the array
- Use the square brackets [] to indicate that the parameter is an array

Ex) this method has a parameter that is an array of integers called numbers

```
public static void tryChanging(int[] numbers) {  
    numbers[1] = 123;    changes the 2nd element in the array to 123  
}
```

and here is a call:

```
int[] myNums = {5, 8, 3};  
System.out.println(myNums[1] + " " + myNums[0]);  
tryChanging(myNums, myNums[0]);  
System.out.println(myNums[1] + " " + myNums[0]);
```

Output:
8 5
123 5

An array is a *reference data type* – so passing a whole array to a method passes the reference to the elements, allowing the method to access an element of the array and change its value.

Arrays with Meaningful Indexes

- Many algorithms make use of the index value of an array element for simplifying the storage and retrieval of data
- For example, a testScores array with 101 elements indexed from 0 to 100 could store a count of all the scores of 90 in the element with index 90, the count of scores 82 in element 82, and so on.

Programming Exercise:

Create a StudentRoster application that prompts the user for the number of students in the class and then prompts the user for each student's name and stores the names in an array. After all the names have been entered, the application should display the title "Student Roster" and then list the names in the array.

Submit your source code to the Google Doc "ICS4U – Activity Submission Form"