Lesson 10-1: Arrays

Computer Science 46A: Introduction to Programming

San José State University

Announcements

- Lab tomorrow
- Next weeks lecture by another professor – Professor Vidya Rangasayee
- When is Midterm 2?
- When do we have the final exam?

Learning Objectives

By the end of this lesson, you should be able to:

1. Create arrays that store Objects from a given class
2. Use methods from the Array class to modify arrays
3. Shuffle array elements to “add” or “remove” values from an array

Arrays with Objects

Arrays Revisted

- To declare an array, use the following syntax:

```
[type][] [name] = new [type][size]
```

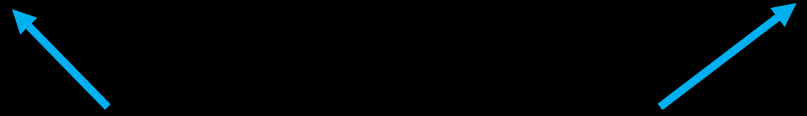
- arrays don’t have methods - we access the elements by referencing their indices in square brackets []
- Once initialized, arrays don’t change size
- array Example:

```
int[] myNumbers = {1, 2, 3, 4, 5}
```

Arrays with Objects – an example

- arrays can store objects from any class (in addition to primitive types)
- Syntax:

```
[type][] [name] = new [type][size]
```



Class Name of Objects in array

- Example:

```
String[] [name] = new String[size]
```

Example: CoffeeArrayProg

Coffee class stores roast and size

CoffeeArrayProg is a class that creates an array to manage an array of different **Coffee** objects

See example code in CoffeeArrayProg

Methods from the Array Class

Array Class Methods

- Last time, we saw that we can import the Array class to print arrays
- The Array class also has a variety of other methods, e.g.:
  - binarySearch
  - copyOf
  - fill
  - sort
- The complete list of methods are listed in the API HERE

Array Class Methods: binarySearch and sort

- `binarySearch(array[], value)`
  - Search the array and return the index of value
    - If the value is present, return the index
    - If not, then return the value where it would be present (as a negative number)
  - Note: the binarySearch method only works if the array is sorted!
- `sort(array[])`
  - Sort the array in ascending/numerical order

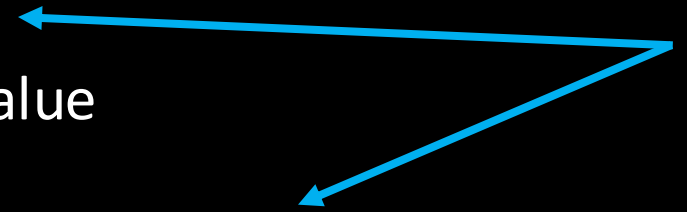
See example code in ArrayMethods

Array Class Methods: copyOf and fill

- `copyOf(array[], newLength)`
  - Make a copy of the array
    - If the newLength is longer, pad the array
    - If the newLength is shorter, then truncate the array

- `fill(array[], value)`
  - Fill the array with the given value

- `fill(array[], startIndex, endIndex, value)`
  - Fill the array with the given value



See example code in ArrayMethods

Poll Everywhere: Question 1

What kind of method is Arrays.sort()?

A) mutator

B) accessor

C) neither

Code

```
char[] letters =
            {'M','i','k','e','W','o','o','d'};
System.out.println("unsorted letters = "
                   + Arrays.toString(letters));
Arrays.sort(letters);
System.out.println("sorted letters = "
                    + Arrays.toString(letters));
```

Output

```
unsorted letters = [M, i, k, e, W, o, o, d]
sorted letters = [M, W, d, e, i, k, o, o]
```

PollEv.com/narayanbalasubramanian644

Arrays as Instance Variables

Adding and removing elements

Consider the following array called letters:

0	1	2	3	4	5	6	7	8	9
A	M	i	k	e	W	o	o	d	

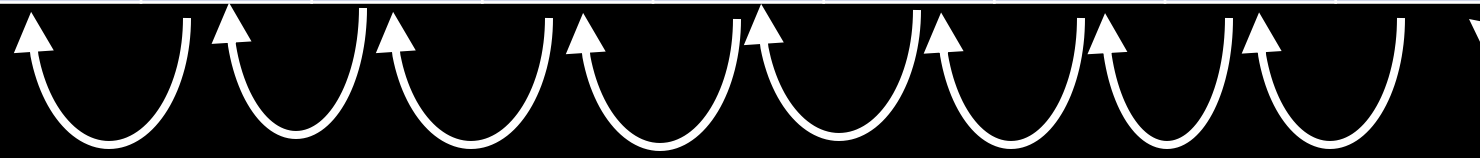
the `letters` array

Code to generate letters:

```
char[] letters = new char[10];
letters[0] = 'A';
letters[1] = 'M';
letters[2] = 'i';
letters[3] = 'k';
letters[4] = 'e';
letters[5] = 'W';
letters[6] = 'o';
letters[7] = 'o';
letters[8] = 'd';
```

“Removing” an element at index 0 from an array

0	1	2	3	4	5	6	7	8	9
A	M	i	k	e	W	o	o	d	



Step 1: Shuffle all letters to the left, starting with the first element

Step 2: Replace the last value with the default array value

Result:

0	1	2	3	4	5	6	7	8	9
M	i	k	e	W	o	o	d		

See example code in AddingRemovingChars

Poll Everywhere: Question 2

What is the length of the `letters` array?

0	1	2	3	4	5	6	7	8	9
M	i	k	e	W	o	o	d		

A) 7
B) 8
C) 9
D) 10

PollEv.com/narayanbalasubramanian644

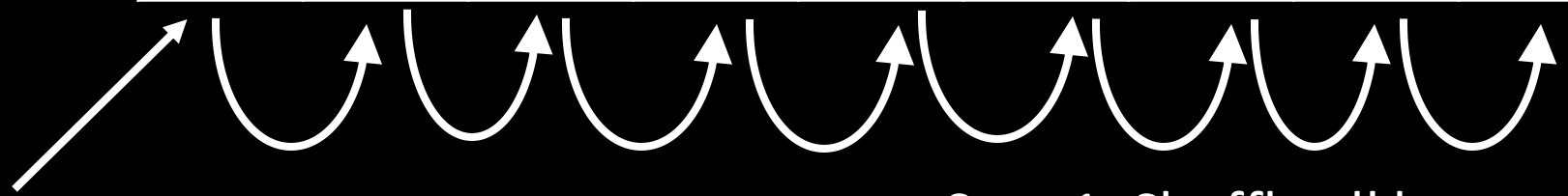
Some Default Array Values

When arrays are initialized, all values are filled with default values according to the type of array:

Type	Default Value
int, long	0
float, double	0.0
boolean	false
char	‘ ‘
String, Object	null

“Adding” an element at index 0 into an array

0	1	2	3	4	5	6	7	8	9
M	i	k	e	W	o	o	d		



Step 2: Replace the first value with the new value

Step 1: Shuffle all letters to the right, starting with the last element

Result:

0	1	2	3	4	5	6	7	8	9
B	M	i	k	e	W	o	o	d	

See example code in AddingRemovingChars

Participation Exercise 10-1a: FrogArrayProg

Goals:

1. Create an array that stores Frog objects generated randomly
2. Write a method that will swap two objects in the array

```
Enter the number of frogs: 3
Enter an integer as the random generator seed: 133
All frogs before swapping:
Frog[Weight:18,Legs:2]
Frog[Weight:11,Legs:2]
Frog[Weight:23,Legs:4]
[Frog[Weight:18,Legs:2], Frog[Weight:11,Legs:2], Frog[Weight:23,Legs:4]]
All frogs after swapping:
Frog[Weight:23,Legs:4]
Frog[Weight:11,Legs:2]
Frog[Weight:18,Legs:2]
[Frog[Weight:23,Legs:4], Frog[Weight:11,Legs:2], Frog[Weight:18,Legs:2]]
```

Output of the FrogArrayProg class

Codecheck Link: HERE and on Canvas

Participation Exercise 10-1b: PersonArray

Goals:

1. Create an array with 3 Person objects
2. Add a new Person object to the array at a given index
3. Remove an object from the array at a given index

```
Array contents before modification:
    Person[initials=MW,age=17]
    Person[initials=QC,age=81]
    Person[initials=BO,age=43]

Array contents after addition:
    Person[initials=MW,age=17]
    Person[initials=GW,age=61]
    Person[initials=QC,age=81]
    Person[initials=BO,age=43]

Array contents after removal:
    Person[initials=MW,age=17]
    Person[initials=GW,age=61]
    Person[initials=BO,age=43]
```

Output of the PersonArray class

Codecheck Link: HERE and on Canvas