Topics: One-dimensional array

Reading (JV): Sec 8.1-8.4; Sec 8.5.1, 8.5.2, 8.5.4

Arrays

- Arrays are objects. An array is an ordered list of values (or objects) of one type
- The entire array has one name (identifier)
- Each element in the array has an integer index (begins at 0)
- An array of length \( N \) is indexed from 0 to \( N-1 \)

Array declaration and construction

- Declaration syntax: \( \text{type}[] \text{identifier}; \)
  Examples:
  
  ```
  int[] counts;
  double[] price;
  String[] names;
  Interval[] series; // assuming an Interval class has been defined
  ```

- Instantiation syntax: \( \text{new type[ size ]} \)
  Example:
  ```
  new int[4]
  ```

- Declaration and instantiation
  ```
  int limit= 4;
  double[] price; // declaration
  price= new double[limit]; // instantiation and assignment
  ```

- Creating an array using an initializer list

The size of an array is held in the constant \( \text{length} \). \( \text{length} \) is automatically defined when an array is created and cannot be changed. In the above example, the expression \( \text{price.length} \) gives the size of the array \( \text{price} \).

Index operator \( [\] \)

The expression \( \text{identifier[integer_expression]} \) accesses an element in the array referred to by \( \text{identifier} \)

Examples:

```
int[] freq = new int[101]; // declaration & instantiation
freq[9+70] = 17; //set freq[79] to 17 (freq[79] is the 80th element in freq)
int grade = Keyboard.readInt(); //assume the value to be 1 to 100 inclusive
freq[grade] = freq[grade] + 1;
freq[grade]++;
```

In the example above, the expression \( \text{freq[2]} \) represents an integer and can be used anywhere an \( \text{int} \) variable can be used.
Pattern for processing an array

// assume an array has been created and is referred to by variable arr
for (int i=0; i<arr.length; i++) {
    // perform some process (on arr[i])
}

Example

Create an array of length 6 filled with random numbers in the range of 5 to 9. Calculate the sum.

Sorting

- Arrange elements in a list by some specified order
- Sort “in-place” means sort without using extra memory space for holding another copy of the array
- There are many sorting algorithms: selection sort, insertion sort, bubble sort, etc.

Template for selection sort (ascending order)

// Loop from first to second last element

    // Find index of minimum value _____________________________

    // Swap ith element with minimum value