CS115 Week 14
Arrays

Assigning values to array elements

double[] temps = new double[5]; // Creates array
int m = 4;
temps[2] = 98.6;
temps[3] = 101.2;
temps[0] = 99.4;
temps[m] = temps[3] / 2.0;
// What value is assigned?

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>99.4</td>
<td>?</td>
<td>98.6</td>
<td>101.2</td>
<td>50.6</td>
</tr>
</tbody>
</table>

What values are assigned?

double[] temps = new double[5]; // Allocates array
int m;
for (m = 0; m < temps.length; m++)
temps[m] = 100.0 + m * 0.2;

What is length?

|----------|----------|----------|----------|----------|

Variable subscripts

double[] temps = new double[5];
int m = 3;

What is temps[m + 1]?
What is temps[m] + 1?

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>100.0</td>
<td>100.2</td>
<td>100.4</td>
<td>100.6</td>
<td>100.8</td>
</tr>
</tbody>
</table>

Initializer List

int[] ages = {40, 13, 20, 19, 36};
for (int i = 0; i < ages.length; i++)
System.out.println("ages[" + i + "] = " + ages[i]);

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>13</td>
<td>20</td>
<td>19</td>
<td>36</td>
</tr>
</tbody>
</table>

Passing Arrays as Arguments

- In Java an array is a reference type. The address of the first item in the array (the base address) is passed to a method with an array parameter
- The name of the array is a reference variable that contains the base address of the array elements
- The array name dot length returns the number of locations allocated
public static double average(int[] grades)
// Calculates and returns the average grade in an
// array of grades.
// Assumption: All array slots have valid data.
{
    int total = 0;
    for (int i = 0; i < grades.length; i++)
        total = total + grades[i];
    return (double) total / (double) grades.length;
}

Passing an Array as Parameter

Partial Array Processing
• length is the number of slots
  assigned to the array
• What if the array doesn’t have valid
  data in each of these slots?
• Keep a counter of how many slots
  have valid data and use this counter
  when processing the array

More about Array Indexes
• Array indexes can be any integral expression of type
  char, short, byte, or int
• It is the programmer’s responsibility to make sure
  that an array index does not go out of bounds. The
  index must be within the range 0 through the array’s
  length minus 1
• Using an index value outside this range throws an
  ArrayIndexOutOfBoundsException; prevent this
  error by using public instance variable length

String[] groceryItems = new String[10];

String[] groceryItems = new String[10];
groceryItems[0] = "cat food"
groceryItems[1] = "rice"
.. .. ..
groceryItems[8] = "spinach"
groceryItems[9] = "butter"

Date[] bigEvents = new Date[10];

Date[] bigEvents = new Date[10];
bigEvents[0].getMonth() int
bigEvents[0].getYear() int
bigEvents[0].setMonth(3) void