Arrays

Lecture 22

Borrowing from slides by Alan Hu, Kurt Eiselt, Paul Carter, and Tamara Munzner
News

- Assignment 2
  - Due TODAY

- Midterm 2 coming up next week
  - Wednesday, Nov 4, 6:30-7:30
  - Note: the rooms will be different
  - Material: conditionals, loops
    - Arrays will NOT be on the midterm
Reading Assignments

- Reading for this week: arrays
  - Edition 3: Ch. 7.1, 7.5-7.7
  - Edition 2: Ch. 8.1, 8.5-8.7
Objectives for Today

- Learn different strategies for debugging code
- Gain an initial understanding of arrays
Arrays

- Arrays let you create a bunch of variables (all of the same type), that you refer to by array name and number (called the index).

Example:

```java
int[] x = new int[10];
```

instead of

```java
int x0, x1, x2, x3, x4, x5, x6, x7, x8, x9;
```
Declaring Arrays

- Must declare, just like any other variable.
- Declarations look like creating objects:
  ```java
  Scanner s = new Scanner(System.in);
  int[] x = new int[10];
  ```
Declaring Arrays

- Must declare, just like any other variable.
- Declarations look like creating objects:
  
  ```java
  Scanner s = new Scanner(System.in);
  int[] x = new int[10];
  ```

  Type of the variable
Declaring Arrays

- Must declare, just like any other variable.
- Declarations look like creating objects:
  
  ```java
  Scanner s = new Scanner(System.in);
  
  int[] x = new int[10];
  ```

  The variable name. In both cases, these are reference variables.
Declaring Arrays

- Must declare, just like any other variable.
- Declarations look like creating objects:
  
  ```java
  Scanner s = new Scanner(System.in);
  ```

  ```java
  int[] x = new int[10];
  ```

  The keyword `new` to allocate memory
Declaring Arrays

- Must declare, just like any other variable.
- Declarations look like creating objects:
  ```java
  Scanner s = new Scanner(System.in);
  int[] x = new int[10];
  ```

The constructor for the object or array
Declaring Arrays

- Syntax:

  ```java
type[] variableName = new type[size];
  ```

- Examples:

  ```java
double[] dataSet = new double[30];
int[] quizScores = new int[5];
String[][] args;
  ```
Using Arrays

- Syntax:
  
  `variableName[index-expression]`

- If `var` is an array (reference) variable of type `type[]`, then `var[expr]` is a variable of type `type`.

- Examples:

  ```java
double[] dataSet = new double[30];
dataset[0] = dataSet[1]/2;
```
Array Length

- Syntax: \texttt{variableName.length} (note no parentheses)

- If \texttt{var} is an array (reference) variable, then \texttt{var.length} is an \texttt{int} containing number of entries in array.

- Examples:
  
  \begin{verbatim}
  double[] dataSet = new double[30];
  dataSet.length == 30
  \end{verbatim}

- Note that indices go 0 .. \texttt{dataSet.length-1}!
Using Arrays

```java
int[] x = new int[10];
x[0] = 5;
x[1] = 7;
x[9] = x[0] + x[1];
```

![Diagram of an array with elements 5 and 7 at indices 0 and 1, and the result 12 at index 9]
Array Initializers

Array creation and initialization can be combined:

- int[] x = {1,2,3};
- String[] colors = {"red", "green", "blue"};
Example: Student Records

- Design a program that keeps track of students enrolled in CPSC 111:
  - Name (just first name for simplicity)
  - Grade (a single percentage for now)
Student Records

Let’s write an example application using these student records:

- Grade entry
  - Names of students provided on the command line
  - Ask the user to input the corresponding grade for each name
Let’s write an example application using these student records:

- Grade entry
  - Names of students provided on the command line
  - Ask the user to input the corresponding grade for each name

- Grade query
  - Ask the user to type in a name
  - Find the corresponding grade and print it…