Self Test

What's the scope of variable i?

class testScope
{
  public static void main(String[] args)
  {
    for (int i=0; i<10; i++)
      System.out.println(i);
  }
}

In the beginning...

We thought mostly about what we were going to do with data – functions, calculations, etc.

But what about the data?
• How do we structure data to make it easy to “do things” with it?

Easy?

How can we make it easy to access data?
How can we make it quick to put more data in?
How do we make it quick to get data out?
How do we make it quick to find a specific data element?

The Answer

We create different data structures for our data

You will learn about:
• Arrays, Linked Lists, Queues, Stacks, Trees, Hashing, Graphs

Arrays

An array of basic types

int[]

15 16
length = 2

An array of objects

String[]

"moo" String
"bird"

String[]

Addr1 Addr2
length = 2

Memory Diagram

Draw a memory diagram of the following variables at the end of the following statements (assume no objects have been garbage collected):

int[] shoeSize = {5, 8, 7, 10};
String[] names = {"moo chow", "aquiel", "moi");
shoeSize[2] = 3;
names[0] = new String("argh!");
1-D Arrays are like a table with one row

```
int[] simpleArray = {5.5, 5.7, 7.0, 8.2};
```

- The same things that cause syntax errors with basic types will cause them with arrays too.

2-D Arrays are like a table with rows and columns

```
int nums[][] = new int[2][4];
```

- Rows number comes first
2D Arrays

- `int nums[ ][ ] = new int[2][4];`
- The above is really short for:
  - `int nums[ ][ ] = new int[2][1];`
  - `nums[0] = new int[4];`
  - `nums[1] = new int[4];`
- The second dimension can have different numbers of elements

RIT Computer Science Dept.

Multi-dimensional Arrays

- Multi-dimension arrays can be declared by repeating pairs of brackets up to the required dimension.
- The length instance variable holds the size or length of the array:
  - `String[] words = new String[100];`
  - `System.out.println ( words.length );`
  - `int[][] twoD = new int[10][20];`
  - `System.out.println ( twoD[0].length ); // gives 10`
  - `System.out.println ( twoD[0].length ); // gives 20`

RIT Computer Science Dept.

Example

- Code snippet for determining the average of the third row in the array.
  - `int [][] payScale = {{0, 1, 2, 3, 4},
   {0, 1, 2, 3, 4},
   {0, 1, 2, 3, 4}};
  - `double average=0.0, sum=0.0;
  - for (int i=0; i<5; i++)
    - `sum += payScale[2][i];`
  - `average = sum / 5;`

RIT Computer Science Dept.

Problems in arrays

- When arrays are created they are final (their size may not be changed).
- This means that trying to access a value that is bigger than the array, then an out-of-bounds error will be generated.
- This also means that you need to have an idea of the size of an array before using one.

RIT Computer Science Dept.

Searching and Sorting

- Many problems in Computer Science involve searching for an element in a list or sorting a list.
- The easiest way to search is to go through each element (search exhaustively)
- One of the easiest types of sorting is selection sort

RIT Computer Science Dept.

Selection Sorting

- The basic algorithm (ascending sort):
  1. Find the minimum value in the unsorted part of the array
  2. Swap the min. value with the first value in the array
  3. Repeat with the remaining unsorted part of the array

RIT Computer Science Dept.
Write it

- Write the following method:
  ```java
  private int returnMinIndex(int[] array, int startPos, int endPos)
  ```
- The return value is the index of the minimum integer in the array searched.
- startPos is the starting position of the search.
- endPos is the ending position of the search.
- array is the array to find the minimum in.

Another method

- Write the following method:
  ```java
  private void swapNumbers(int[] array, int firstIndex, int secondIndex)
  ```
- firstIndex and secondIndex are the indices in the array to be swapped.
- array is the array in which to do the swapping.

Putting it together

- Write the following method using your previously defined methods:
  ```java
  public void selectionSort(int[] array)
  ```

Readability

- Sometimes using methods can help:
  - In breaking down the complexity of a problem.
  - In writing a solution that is easy to read.

Write it

- Write a class to find the maximum number of an integer array of 5 ages entered at the command line.
- You must create an integer array to hold the numbers.
- In order to convert strings to integers you may use the following method:
  ```java
  int Integer.parseInt(String str)
  ```

One potential answer

```java
class SelfTestArray {
    public static void main(String[] args) {
        int numInArray = args.length;
        int[] numbers = new int[numInArray];
        int maximum;
        for (int i=0; i<numInArray; i++)
            numbers[i] = Integer.parseInt(args[i]);
        maximum = numbers[0];
        for (int i=0; i<numInArray; i++)
            if (maximum < numbers[i])
                maximum = numbers[i];
        System.out.println("The maximum value is: "+ maximum);
    }
}
```