

Arrays in Java

9-19-2002

Opening Discussion

- What did we talk about last class?
- How many of you intend to take Data Abstraction next semester?
- There are no "friends" in Java. Even in C++ they are considered poor programming style.
- The design of classes does include the methods that are in them. They need to be stubbed out and documented.

Keeping Track of Things

- One of the things that we love to be able to in programs is to be able to keep track of the information we are using. That is to say that we want to be able to save off values and pull them back for later use.
- The simplest structure for doing this in Java (and many other languages) is an array. Basically an array gives us a way to refer to things with numeric indices.

Arrays are Objects

- In Java, arrays are objects. For every type, A, in Java, there is a type that represents and array of A that we write as A[].
- As with other objects in Java, what we really declare is a reference to that object and we have to allocate it before we can use it.
- The syntax otherwise is much like in C/C++.
- Because they are objects, arrays know their length. There is a public integer member of the array classes called length that you can read it from.

Arrays of References

- Arrays of primitive types act the same way in Java as in C/C++. However, arrays of objects are really arrays of references to objects.
- This enables polymorphism, but also requires that each object in the array must be allocated individually before it can be used.

Initialized Array Syntax

- When you declare an array in Java you can use special syntax to initialize it. You only want to do this with short arrays.
- The syntax you use is to have a comma separated list of array elements in curly braces like this.

```
int[] a={4,7,2,-9,2};
```

Sorting

- One of the things that we want to be able to do on computers with any sequential collection is to sort the elements of that collection. Here are several "slow" sorts that you should know of.
- Bubble Sort: Repeated passes swapping out of order adjacent elements. Elements "bubble" to far end. $O(n^2)$

More Sorting

- Selection Sort: Each pass you find the min/max of what is left unsorted and swap it to the end. $O(n^2)$
- Insertion Sort: Walk through each elements, inserting it into the earlier elements. $O(n^2)$
- Optional: Shell's Sort: Repeatedly does insertion sort on subsets made with "diminishing gaps". $O(n^{3/2})$ or so.

Searching

- The other activity we are typically interested in doing with collections is sorting for items in them.
- With a general collection all we can do is a sequential search where we walk through all the elements until we find what we want. Obviously, if you do this frequently, it can be a bit slow. $O(n)$

Fast Searching

- If you have a sorted array you have other options. The most common of which is a binary search.
- Start by looking in the middle of the array. If it is what you are looking for then you are done. If it is bigger than what you want you only have to look on the bigger side and the opposite is true for smaller values. Repeat this always cutting the size in half. $O(\log(n))$

java.util.Arrays

- This is a "Utility" class that has only static methods in it. All the methods in it work with arrays.
- It includes methods for sorting and searching. While you probably won't be using this directly in this class, it is a very good thing to know about because it can make your work faster in the future.

Multidimensional Arrays

- A natural extension of arrays is to have arrays of several dimensions. In Java this is accomplished by using the potentially recursive definition of arrays.
- Remember I said that any type can be used with an array and the array of the type is a type itself. This means that the type in an array can be an array type.
- Basically you can put as many [] after a type as you want.

Allocating Multidimensional Arrays

- As with any array of objects, allocating the array doesn't allocate the elements in the array, just space for references to them. You need to allocate the arrays in the array yourself.
- If you are allocating a "rectangular array" Java has an abbreviated syntax where you can specify the size of many dimensions in one new statement.

Your Screen Class

- You are going to need to use a 2D array in your screen class to hold the blocks that make up the screen.
- The blocks can be accessed by x,y coordinates placed in the normal way in computer graphics with 0,0 at the top left.
- Let's also revisit the two methods that help the screen editor interact with the blocks on your screen.

Let's finish up our code

- Now we will try to finally finish up the code that we have done the last two classes then maybe start something new that involves some arrays. I'm open to suggestions if you can think of anything.

Minute Essay

- Write code to declare two 2D arrays of type Shape. Make the first a square of 6x6. Make the second a triangle with the length of the "second" array equal to the first index plus one.
- Remember to turn in the design for assignment #2 to me by this evening. Also check the description of the assignment again because I have put more information and code up.
