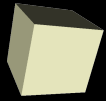




Sorting and Searching

2/11/2008





Opening Discussion

- Do you have any questions about the quiz?
- Let's look at solutions to the interclass problem.
- Do you have any questions about the reading?
- Do you have any questions about the assignment?
- Assignment submissions
 - ◆ I have graded assignment #1. At least as much as people submitted.
 - ◆ Don't submit CVS files. I need regular .java file. You can select the full project directory in your workspace and submit that.
 - ◆ Comments are posted about game ideas.



Sorting and Searching Arrays

- These are topics that you should have talked about a fair bit in PAD1 so I'm not going to lecture on them much now.
- Instead, we'll write some code that uses arrays and these concepts.

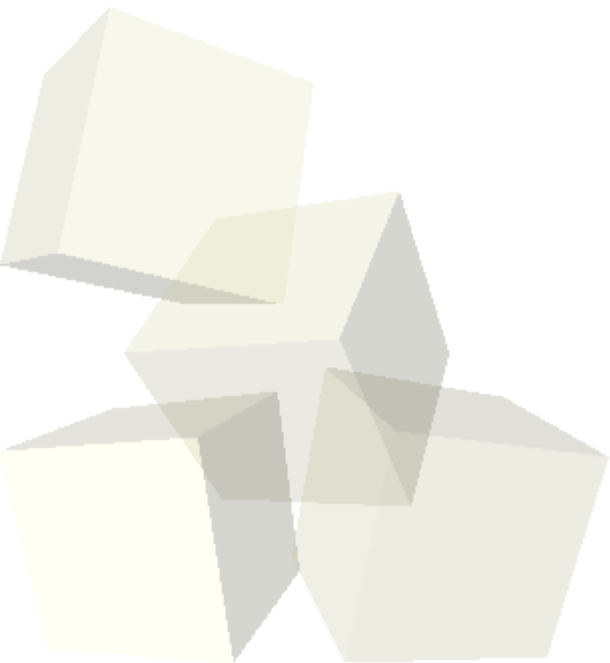




- A function $g(n)$ is $O(f(n))$ iff

$$\exists n, c : \forall m > n, c * f(m) > g(m)$$

- Let's look at what this means graphically.



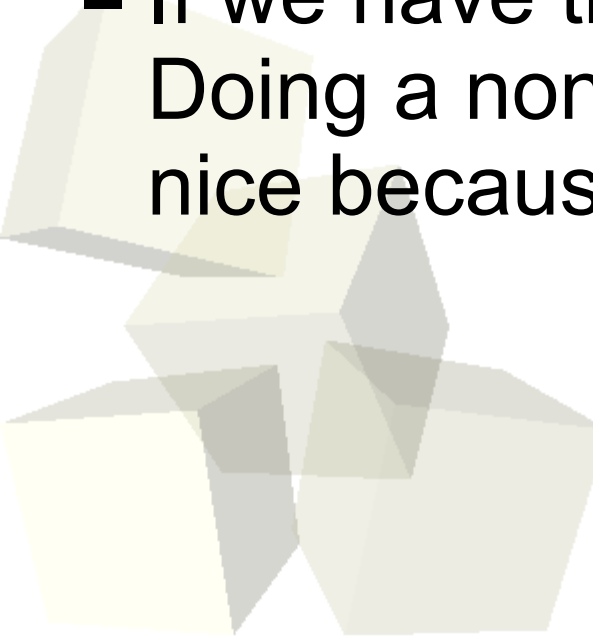


Polymorphic Sorts

- One of my motivating examples for polymorphism was a sort. In C you have to write a separate sort for every type, or you have to do some very odd stuff. In Java we can write polymorphic sorts of object types in at least two ways.
- You can write a sort/search that only takes subtypes of Comparable.
- You can write a sort/search that works on any Object, but that also takes an object of type Comparator.
- I prefer the second method as it is far more flexible.
- The `java.util.Arrays` class contains some utility methods.



- Let's write a method that uses one of the sorts you know to sort any object type. Try to make this a generic method so that it will be type safe. You can put it in a class called `ArrayHelper`.
- Let's make it so our comparator counts how many comparisons are made so we can see what sorts are best.
- If we have time we can write a search as well. Doing a non-recursive binary search would be nice because your book uses a recursive one.





Adding to the Ray Tracer

- Arrays are a big advancement for us in our programming power because they allow us to keep track of multiple things.





- What sorts do you remember from PAD1?
Explain how one of those sorts works.
- Remember that the design for assignment #2 is due on Thursday.
- Interclass Problem – Write a polymorphic quicksort that uses a comparator.

