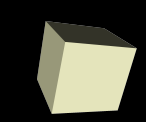
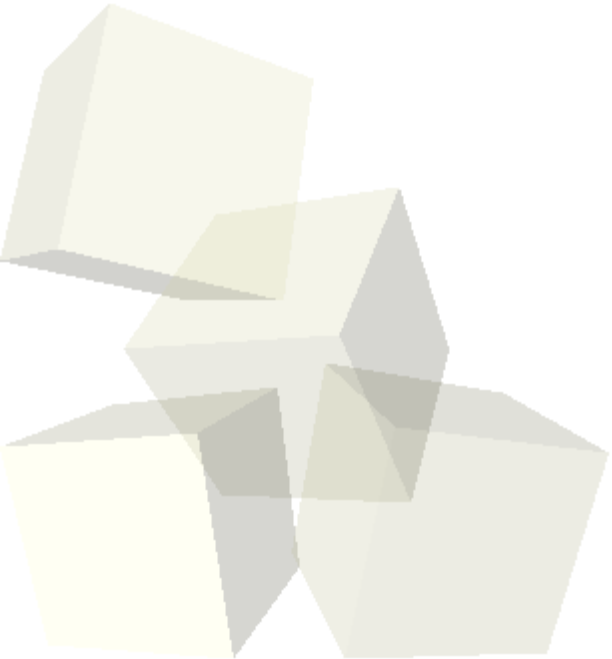
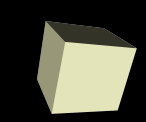


Typedef, enums, and structs

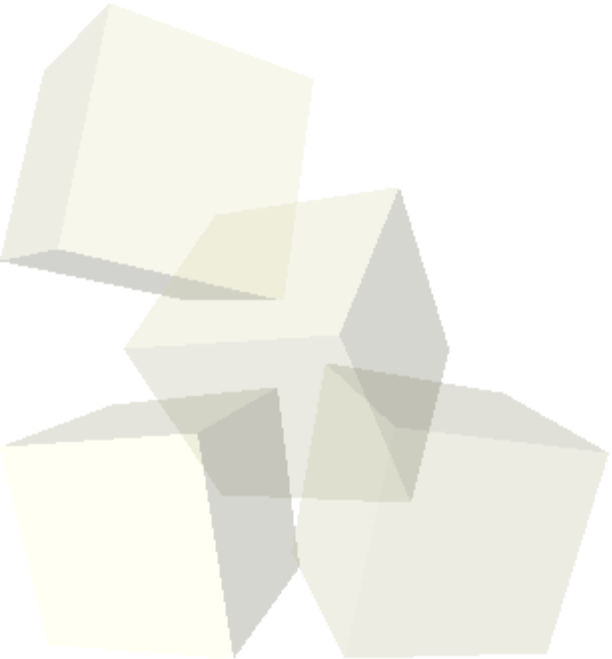
11/7/2007





Opening Discussion

- Let's look at solutions to the interclass problem.
- Do you have any questions about the assignment?



User Defined Types

- So far we have only worked with types that the C language provides for us.
- Pointer types and array types allow us flexibility, but they are still built in types.
- Virtually all modern programming languages allow user defined types. These are types that the programmer can add in.
- User defined types don't provide any fundamental power, but they make it much easier to organize your code.

typedef

- The typedef statement simply gives a new name to a type. This can be done with any type.
- Doing this with primitives can make code easier to move across platforms.
- The syntax is:
  - typedef existingType newName;
- The type size_t we have seen in the man pages is just a typedef for some integer type.
- Some software projects like to define types like int32 and uint32 among others.

Enumerations

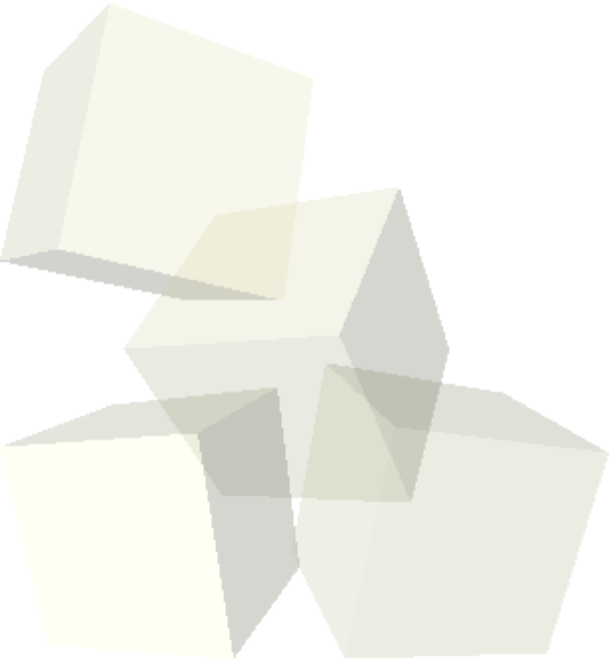
- An enumeration is a way of defining a small set of names of a particular type. A classic example is the colors of a street light.
  - enum LightColor { red, yellow, green };
- The names in the enum are actually just bound to integer values. In this case, 0, 1, and 2. You can also specify the value you want an enum to have.
- Enumerations are very often used along with switch statements.
- The primary downfall of enums in C is that they aren't really typesafe. They are just ints when you get down to it.

Structures

- More interesting user defined types can be created with structures. A structure is a grouping of multiple other types.
- They are typically declared in one of two ways.
  - Tagged
    - struct Name {
      - Fields
    - };
  - With typedef
    - typedef struct {
      - Fields
    - } Name;
- Your book pushes for names with all caps. I'm going to use the Camel naming starting with a cap.

Using Structures

- When we declare an instance of a struct we can get to the individual parts using dot notation.
  - room.desc
- If you have a pointer to a struct there is a shorthand you can use with an arrow.
  - (*room).desc
  - room->desc



Minute Essay

- Write a struct for a student.
- Interclass Problem – Write a program in which you make a struct that contains an array of ints. Write a function to sort it then print the contents back in main. You might need to play with this some to make it work.