

# CS350: Data Structures Class Notes

*Class 3: Friday, September 5, 2008*

## **Announcements:**

Shay Ellison will be tutoring in Olin 207 Sunday thru Thursday from 8:00 to 10:00PM.

## **Abstract Data Types Continued**

Last time we did an intuitive introduction to ADT saying ADT's have

- Specifications—what it does
- Implementations—how it does it

It is more correct to say the ADT *is* the specification and an ADT *has* an implementation.

Specifications include data and operations. We may need to define additional notation. See the Shiflet example for an ADT for Integers.

Example: ADT for a LIST

Data: items on list

Operations (possible) include: create, destroy, empty?, size, first, rest, insert, delete, view

The first four (create, destroy, empty?, & size) are very common for aggregate data structures. Some (e.g., insert, delete, view) can take a position in the list as an argument, can default to the first item in the list, or can have an implied position (See Shiflet example distributed in class for the latter.)

Notice that some operations can be defined in terms of others. E.g.,

list.view() could be written as

```
x = list.remove()
list = list.insert(x)
return x
```

Consider LISP/SCHEME with NIL, CAR, CDR, CONS, EMPTY? These are enough to define the List type (along with implicit operations such as garbage collection).

Obviously, we have a lot of choices on what function to include in an ADT

Example of a **UML (Uniform Modeling Language) Diagram**:

LIST
items
create() destroy() empty() size() insert() remove() retrieve()

**Axioms:** We can formally describe the properties of an ADT. For example:

1. When creating a list, its size is 0
2. When creating a list, it is empty
3. After inserting into a list, its size increases by 1
4. After inserting into a list, its size is greater than 0
5. After inserting into a list, it is not empty
6. If we insert and then delete, we have the original list
7. When creating a list, if we remove an item we generate an error
8. When creating a list, if we retrieve an item we generate an error

### **Pre/Post Conditions**

These should be clearer at this point:

For `x = list.retrieve()`

Precondition: the list is not empty

Postcondition: `x` is the first item in the list and the list is unchanged.

(Of course, this will depend on exactly what you mean by `retrieve()` and could be different in a different ADT.)

Homework: Use the examples by Shiflet as a model to create an ADT for **BOOLEANS**.