

## Homework #3 Data Structures

Due: February 20, 2010 (Saturday at 11:59 PM)

### To-Do List via Unsorted Doubly-Linked List Class

Design an interactive To-Do list program that uses a templated unsorted doubly-linked list class to store the To-Do list items. Design and test the unsorted doubly-linked list class separately from the To-Do list application.

The list class should support the following operations:

(constructor) Construct list  
(destructor) List destructor  
operator= Copy container content

#### Iterators:

begin Return iterator to beginning  
end Return iterator to end  
rbegin Return reverse iterator to reverse beginning  
rend Return reverse iterator to reverse end

#### Capacity:

empty Test whether container is empty  
size Return size

#### Modifiers:

push\_front Insert element at beginning  
pop\_front Delete first element  
push\_back Add element at the end  
pop\_back Delete last element  
insert Insert element before a position specified by an iterator  
(e.g., myList.insert ( myIter, newValue))  
erase Removes from the list either a single element pointed at by an iterator or a range of elements specified by two iterators ([firstIter,lastIter)).  
swap (optional) swap contents between two lists  
clear Clear content removes all items from the list  
remove Remove all elements with specific value

The corresponding iterator class for the list should handle operators for ++, --, \* (dereferencing for input and output), = (assignment), == and !=.

I'll leave the To-Do list specifications vague except that you should be able to create an empty To-Do list or load a To-Do list from a file, and save a To-Do list to a file. You should also be able to interactively add items anywhere, delete items by position or value, view items, etc.

**Submit your homework electronically at [http://www.cs.uni.edu/~schafer/submit/which\\_course.cgi](http://www.cs.uni.edu/~schafer/submit/which_course.cgi)**

The steps for the homework submission system are:

1. Write, debug, and test your program. Zip all files together in a file called Save it in a file called todo.zip
2. Log on to the submission system at: [http://www.cs.uni.edu/~schafer/submit/which\\_course.cgi](http://www.cs.uni.edu/~schafer/submit/which_course.cgi)

(It is very likely that you will get some security certificate warnings when trying to use this. You may add an exception and accept the existing security certificate.) Use the same AD-ITS User name and password you use to log on the lab computers.

3. Select the course and section number of "810:052, Data Structures, Fienup". Click the "Continue" button.
4. Select the homework that you wish to submit: "HW 3 To-Do List". Click the "Continue" button.
5. Specify how many extra files you want to submit. Just leave it at 0. Click the "Continue" button.
6. Upload your program by Browsing and selecting your todo.zip file. Click the "Continue" button.
7. The next page reports on the status of the upload(s). You can always continue to upload a better version of the program until the deadline. The newer file will replace an older file of the same name.