UNBC Laboratory Assignment
Computer Science 200—Fall 2007

Lists and Merge Sorting

Purpose:

The primary purpose of this assignment is to learn how to use the list surgery functions for the STL `list` class.

Due Date:

This assignment is *not* due. It is for practise only.

Assignment:

1. Implement a random shuffle algorithm that works directly on `list`s, and that runs in $o(n^2)$ time.
   (a) Use list surgery (`splice`) to move all of the elements, rather than using functions that would actually cause the elements to be copied.
   (b) Make your algorithms templated on the kind of data that it can receive.
   (c) Use the merge sort algorithm, except that instead using an order to drive the merge stage, make random choices. In order to obtain random lists where every permutation is equally likely, when choosing between the element at the head of a list of size m and the element at the head of a list of size n, you should choose the former with probability $m/(m+n)$.

   BONUS Prove that this is the correct probability.
2. Verify that for short lists, your algorithm gives all possible permutations with equal probability.
3. Verify that for long lists that the original item one has equal probability of ending up anywhere in the list.
4. Time your randomizing procedures for various different sized data collections.
5. Graph your data to verify the running time is $o(n^2)$ behaviour.

[This page solely to demonstrate that I can get the headers right.]