

Data Structures and Algorithms  
**Assignment #2**

Due: Friday January 24

For all these exercises use pseudocode.

1. (10 points) Weiss Problem 3.2

For part (a), assume that you are given pointers to a node  $p$  and its predecessor, while for part (b) you are only given a pointer to a node  $p$ . In both cases, you swap  $p$  and its successor.

2. (10 points) Weiss Problem 3.17

The list has  $N$  nodes. For part (a) you can use  $O(N)$  extra space and you have the choice of using a recursive or a non-recursive method. For part (b) you must use only  $O(1)$  space. Recall that recursive routines can be deceptive in the amount of space they require. For example, for each level of recursion you need to store at least the return address and therefore if you have  $N$  levels of recursion you need  $O(N)$  space.

3. (5 points) Weiss Problem 3.19 (b). You should write methods such as `InsertatFront`, `FindandDelete` (not necessarily these two but something roughly equivalent).

4. (12 points) Weiss Problem 3.24

The array has size `MaxSize`. It only contains elements of type, say, `real` (i.e., it does not contain indices). The routines to implement are `Push`, `Pop`, `IsEmpty`, and `IsFull`. One of the arguments to the `Push`, `Pop`, and `IsEmpty` routines is the number of the stack (0 or 1). On the other hand `IsFull` does not require any argument.

Could you use the same technique if you had 3 stacks? Justify your answer.

5. (10 points) Weiss Problem 3.35

For part (a), your extra space can be a field in each node for “marking” purposes. Part (b) is more challenging! Try and convince yourself that your solution is correct.