Due: Sunday, January 21, 2006 at 11: 59 PM (thirteen-hour grace period)
Instructions: Hand all of your solutions in on paper.

1. On an IA32 machine (e.g., lin01.ugrad.cs.ubc.ca), using gcc, compile the following C program with the -O2 option and the -S option to generate assembly code. Carefully comment every line of the resulting assembly code to explain what that line does. You can ignore the instructions at the beginning and end that play with %ebp and look something like this:

   ```
   pushl %ebp
   movl %esp, %ebp

   ...

   movl %ebp, %esp
   popl %ebp
   ```

   Your comments should relate the assembly code back to specific lines of the C code.

   ```
   int counter = 0;
   int swap(int *a, int i, int j)
   {
     int t = a[i];
     a[i] = a[j];
     a[j] = t;

     ++counter;
     return counter;
   }
   ```

2. Compile swap with the -c option to generate an object file swap.o. Then write a simple test program that calls swap to swap two elements of a two element array of ints. Compile this program along with swap.o to generate an executable. Now, use objdump to disassemble BOTH swap.o and the executable. Include both outputs in your answer and explain the differences in the swap procedure between the object file and the executable.

3. The `movl` and `leal` instructions cause a lot of confusion for new assembly language programmers (like 313 students).

  (a) Explain the fundamental difference between these two instructions.

  (b) Both of these instructions are legal, explain the difference in what they do:

  - ```
    leal    -12(%ebp), %ebx
    ```
  - ```
    movl    -12(%ebp), %ebx
    ```

  (c) Only one of these instructions is legal, choose the legal one, explain why it is legal, explain why the other one is illegal, and explain what the legal one does:

  - ```
    leal    %ebp, (%ebx)
    ```
  - ```
    movl    %ebp, (%ebx)
    ```

  (d) Only one of these instructions is legal, choose the legal one, explain why it is legal, explain why the other one is illegal, and explain what the legal one does:

  - ```
    leal    %ebp, %ebx
    ```
  - ```
    movl    %ebp, %ebx
    ```

4. Textbook problem 3.31