This assignment may be done individually, or by a pair of students. Prepare your answers to the following questions in a plain ASCII text file. Submissions in any other format will not be graded. Submit your file to the Curator system by the posted deadline for this assignment. If you worked in a pair, only one of you should make a submission, but the file should contain the names and PIDs of both students. No late submissions will be accepted.

You will submit your answers to the Curator System (www.cs.vt.edu/curator) under the heading OOC08.

For each question, design an algorithm that satisfies the stated requirements. Express your answer using the pseudo-code notation covered in the course notes on Algorithms. Use descriptive names for your variables, and include comments as necessary. Note: if you do not use the pseudo-code notation form the course notes, we will not grade your submission.

1. [70 points] Suppose you are given a list of N values, each of which is either a 0 or a 1, initially arranged in random value. You need to modify the values in the list so that it consists of a sequence of 0s (possibly empty) followed by a sequence of 1s (also possibly empty), with the same number of both as were originally in the list. For example:

0111010010 → 0000011111

1000111000 → 0000001111

0000000000 → 0000000000

Now this problem could be solved by any of the common sorting algorithms, but the special nature of the values in the list makes it possible to devise a particularly efficient solution. (Here, efficiency would refer partly to how many times you need to reset a value in the list, and partly to how many times you would have to change list position variables in your algorithm.)

Design an efficient solution by completing the following algorithm:

```
#  Sort bi-valued list.
#
number N                 # variable for list size
list number A            # variable for the list of values

get N                    # N = number of values in the list
get A                    # get values for the list

   #  This part is up to you; you may use as many variables as you like,
   #  and whatever seems to you to be the best algorithm (as described in
   #  the comments above.  Note that part of the score will depend on how
   #  Efficient your solution is.

halt                     # done!
```

2. [30 points] Consider an isomorph of the problem given above. Suppose that a collection of N disks are laid out on the ground in a row. Each disk is black on one side and white on the other, but the disks were simply placed randomly. You and a friend are assigned the task of manipulating the disks so that there is a sequence of white-side-up disks (possibly empty), followed by a sequence of black-side-up disks (also possibly empty), with the same number of white-side-up and black-side-up disks as there were to begin with. Write a paragraph describing how you would do this, with the goal of minimizing the amount of work you and your friend must do.