Com Sci 221 — Programming Languages
Assignment #7 — Winter 1994
Due Wednesday, 2 March

1. Given an array $A[bottom..top]$ of integers and an integer v, the *three-way partitioning problem* is to reorder the entries in the array into three ranges: $A[bottom\ ..\ low-1]$ containing values less than v, $A[low\ ..\ high]$ containing values equal to v, and $A[high+1\ ..\ top]$ containing values greater than v. This is sometimes called the *Dutch national flag problem,* because it can be viewed as sorting three colors into the three stripes of the flag. Write an elegant program fragment, using only assignments, conditionals, and **while**-loops, to solve the three-way partitioning problem. Look at problems 3.13 and 3.14 on p. 114 of the text for inspiration. Give preassertion, postassertion, and loop invariant(s) sufficient to show the correctness of your program fragment. You may use whatever mixture of mathematical notation, pictures, and English you find most convenient, but your assertions must refer *only* to the memory state of the computation at a single control point, and must be precise, unambiguous, and sufficiently strong. *This is a pencil and paper problem only, using assignments, conditional, and while: you are* not *required to write an executable program.*

2. Demonstrate that Scheme uses static scope by running a program that would produce different results under dynamic and static scope rules. Example 11.1 on pp. 401–408 of the text shows how to accomplish this—you may merely program this example carefully in Scheme. Do a hand calculation for the same program from McCarthy's equations for the *eval* function to show that the original LISP has dynamic scope.