Assignment 1

April 2, 1996

1 Due Date

April 9, 1996. Please turn in your solution at the beginning of the lecture.

2 Running Scheme

There are several Scheme implementations around. The most robust one is Chez Scheme in `/local/apps/chez-5.0a/bin/chez`. I recommend that you use it. The FAQ of the newsgroup `comp.lang.scheme` lists a number of free Scheme implementations that you can download on your PCs. The CS machines also have scheme48 in `/local/bin/scheme48`.

Start interacting with Scheme *today*.

3 Understanding Recursion

The Towers of Hanoi problem consists of three pegs A, B, and C, and n squares of varying size. Initially the squares are stacked on peg A in order of decreasing size, the largest square on the bottom. The problem is to move the squares from peg A to peg B one at a time in such a way that no square is ever placed on a smaller square. Peg C may be used for temporary storage of squares.

- Write a recursive program in your favorite language to solve this problem. Your program should take n as an argument and should output the sequence of moves. For example, here is a sample run of my Scheme solution:

  ```
  > (hanoi 3)
  move from a to b
  move from a to c
  move from b to c
  move from a to b
  move from c to a
  move from c to b
  move from a to b
  ```

- **Puzzle:** Write a non-recursive solution in Fortran 77. You may use any other language as long as you don't make any recursive calls, pass any functions as arguments, or use any heap allocation. **Hint:** The obvious solution is to implement recursion by maintaining a big array that represents the stack.

4 Object-Oriented Languages

Luca Cardelli is visiting us. Go to his talk this Thursday!