Assessed Exercise – Part II (General)
Introduction to Functional Programming

A Dawar

Issued: 1 February 2000
Due: 3 March 2000

Under the conventional labelling of squares on a chess board, the vertical columns are labelled by the letters A-H from left to right, and the rows are labelled 1-8 from bottom to top. Thus, the bottom left hand corner square (as seen by the player playing white) would be A1, and so on.

1. Define a type `position` suitable for representing a chessboard square.

2. A knight on a chessboard can move to any square that is different from its current position by either 2 rows and 1 column or 2 columns and one row. Write a function `knightsmove` which takes a position and yields a list of positions to which a knight can move from the given one.

3. A knight on an empty chessboard can start at any position, and in exactly 64 moves return to its starting position, having visited every square on the board. This is known as a *knight's tour*. Write a function which given an initial position, will find a tour starting from that position. (**Note**: this involves searching a potentially *very, very* large tree. Think carefully about search strategies.)

Your work should be handed in to Jenni Cartwright in A201 by 12:00 noon on Friday 3 March. Include a program listing and a transcript showing its execution. You should also include a brief but clear explanation of how your program works. Don't forget to include your name, College and user id!