

LISP PROJECT

Instructions: Choose from I-IV below. You may do **ONE** additional problem for extra credit.

- I. Write a LISP program to play "connect-the-dots" with at least a 5 x 5 array. For example user "A" vs computer "C" so far:

```

    .-.-.-.-.-.-.-.
    | A | A | A | ._.
    . . . . | C |
    . . . . . .
    . . . . . .
    . . . . . .
    . . . . . .
    
```

- II. Write a LISP program to solve the "jump-pegs" puzzle found at the Cracker Barrel restaurants (I-75 & SR 24). The object of the puzzle is to leave one peg. The starting position may be selected at random (i.e., pull any peg to start with) and the ending position is as few pegs as possible (minimum 1 peg) anywhere on the board. (You may fix the starting position if you wish. Is fixing the starting position really necessary?). For example:

```

          p
         p  p
        p  p  p   ==>  _ _ _
       p  p  _  p   _  p  _  _
      p  p  p  p  p  _  _  _  _
    
```

You Are a Genius!*?

- III. The minimax and the α - β algorithm are methods for searching game trees. Implement the minimax and the α - β algorithm in LISP. Test your code with examples from our text and others.
- IV. Write a general LISP program to test the Grid-World Robots of Chapters 2, 5 in Nilsson. Inputs include: (1) the size of the grid, (2) the number and location of boundaries, (3) the sensory input and feature vector set and (4) the production rule set used to simulate robot behavior. (a) Use your program to generate the robot motion for the wall-following system on page 28, (b) The system on page 73, (c) Solve Problem 3 in Exam 1, (d) Specify and test a production system to do obstacle avoidance with (1) a robot whose sensor set is $\{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8\}$, (2) a robot whose sensors are defective and can only sense $\{s_1, s_3, s_5, s_7\}$ and (3) a robot whose sensors are defective and can only sense $\{s_2, s_4, s_6, s_8\}$ and (e) Repeat part (d) for a robot that does obstacle contouring.