# Section 1 – Dynamic Programming Pseudocode

Recursive\_Search(State)

Best payoff = payoff(state)

Best action list = []

Best final state = state

For each child action in all valid actions

Child state = result(state, child action)

Child payoff, child action list, child final state = Recursive\_Search(child state)

If child payoff is better than best payoff

Best payoff = child payoff

Best action list = child action + child action list

Best final state = child final state

Return best payoff, best action list, best final state

# Section 2 – Branch and Bound Pseudocode

# Section 3 – Testing Methodology

# Section 4 – Performance and Limitations