Problem1

State: Suppose there are totally N node. For node i, j in N, D(i,j) represents the distance between node i and node j, T(i,j) = 1 if the salesman did travel from node i to j else T(i,j) = 0.

Initial State: Any given starting point, with known distance matrix D, and T(i,j) = 0 for each pair of i,j.

Goal Test: Find a least cost path that starts and ends at the starting node and goes through each other node in the graph once and exactly once. In math, that's minimize $\sum_i^n \sum_j^n D(i,j) * A(i,j)$ while $\sum_j^n A(i,j) = 1, for\ each\ i \in N$

Operators: move to one adjacent node j from current node i, that's set T(i,j) = 1

Operator Cost: The number of the label, or length.

Problem2

Variables: the variables are each row r_i and each column c_i.

Domains: List of words in the dictionary.

Constraints:

1. For each word w, if it is placed horizontally in row r_i, then length of w should be equal to length of r_i; else it is placed vertically, length of w should be equal to length of c_i
2. If a row r_i and a column c_j are intersected, and the point is the m_{th} index of r_i and n_{th} index of c_j, then intersect(i,j) = (m,n), and $r_i[m] = c_j[n]$, means the cross over point letter should be the same.

Problem3

State: [Account Balance (P), Year(Y)]. Initial State is [P, 0], where P means the principle.

Terminate State: if Year = 4, stop.

Action: {Buy CD, Buy Stock}

Transition:

$$T(s, Buy\ CD, s') = 1. if\ s = [P, Y], then\ s' = [1.1P, Y+1]$$

$$T(s, Buy\ Stock, s') = \begin{cases} 0.7, & if\ s = [P, Y], s' = [1.3P, Y+1] \\ 0.3, & \text{if s = [P, Y], s' = [0.9P,Y+1]} \end{cases}$$

Reward:

Reward (s = [P,Y], Buy CD,s'=[1.1P,Y+1]) = 0.1*P

Reward (s = [P,Y], Buy Stock, s' = [1.3P,Y+1]) = 0.3*P

Reward (s = [P, Y], Buy Stock, s' = [0.9P, Y+1]) = -0.1*P

Where P is the current balance.

Set discount factor $\gamma = 1$

$$\pi = \begin{cases} \text{Stock, if Y = 0 OR Y = 1 Or S<1.21} \\ \text{CD, if not the case of buying stock} \end{cases}$$

Problem4

4.1

State: the vector of input sensor, S = [s_1,s_2,s_3,s_4,s_5,s_6,s_7,s_8]
Terminate State: S = [1,1,1,1,1,1,1,1], which means the robot is blocked. Else the robot should run.
Action:{Go North, Go South, Go West, Go East}
Transition: If current action is a legal move, the state will be become the position that the robot moved. Else the action is illegal, the state remains the same. It is impossible to give a transition matrix with probability.

Reward:
Set $x_1 = s_2*s_3$ $x_2 = s_3*s_4$ $x_3 = s_5*s_6$ $x_4 = s_7*s_8$

If $x_4\bar{x}1$, Reward (s, North, s') = 1 else Reward(s,North,s) = -1

If $x_3\bar{x}4$, Reward (s, West, s') = 1 else Reward(s,West,s) = -1

If $x_2\bar{x}3$, Reward (s, South, s') = 1 else Reward(s,South,s) = -1

If $x_1\bar{x}2$, Reward (s, East, s') = 1 else Reward(s,East,s) = -1

If none of above, the robot go north in last condition, Reward(s, North, s') = 0

Policy:

$$\pi = \begin{cases} \text{Go North, if } x4\bar{x}1 \\ \text{Go West, if } x3\bar{x}4 \\ \text{Go South if } x2\bar{x}3 \\ \text{Go East, if } x1\bar{x}2 \\ \text{Go North, Otherwise} \end{cases}$$

4.2
State: the vector of input sensor, S = [s_1,s_2,s_3,s_4,s_5,s_6,s_7,s_8]
Terminate State: S = [1,1,1,1,1,1,1,1], which means the robot is blocked. Else the robot should run.
Action:{Go North, Go South, Go West, Go East}
Policy:

$$\pi(s) = \begin{cases} \arg \max_{a \in \text{Actions}} \hat{Q}(s,a) & \text{with } 1-\epsilon \text{ probability} \\ \text{random } a \in \text{ Actions} & \text{with } \epsilon \text{ probability} \end{cases}$$

Reward:
Set $x_1 = s_2*s_3$ $x_2 = s_3*s_4$ $x_3 = s_5*s_6$ $x_4 = s_7*s_8$

If $x_4\bar{x}1$, if action = Go North, reward = 1;else reward = -1

If $x_3\bar{x}4$, if action = Go West, reward = 1;else reward = -1

If $x_2\bar{x}3$, if action = Go South, reward = 1;else reward = -1

If $x_1\bar{x}2$, ,if action = Go East, reward = 1;else reward = -1

If none of above, the robot go north in last condition, Reward = 0

Then, Use Monte Carlo method to record series of states and actions:

$$D = [s_1, a_1, r_1, s_2, a_2, r_2, s_3, \dots]$$

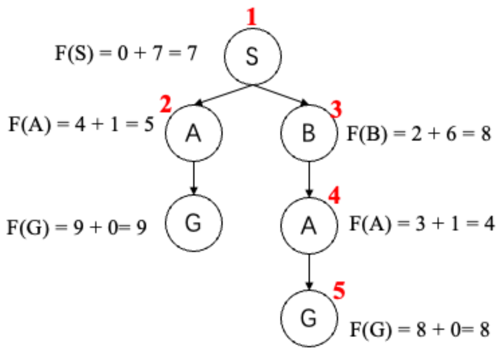
Utility at s_i :

$$u_i = r_i + \gamma r_{i+1} + \gamma^2 r_{i+2} + \cdots$$

Estimate:

$$\hat{Q}_\pi(s,a) = \text{ average of } \{u_t \mid s_t = s, a_t = a\}$$

Problem5



Solution: S -> B -> A -> G

Problem6

I = min(0,7) = 0

F = max(I,5) = 5

G = max(K,L) = 8

C = min(F,G,H) = 4

B = min(D,E) = 3

A = max(B,C) = 4

Problem7

From left to right: Pruned Node N and L

From right to left: Pruned Node K and I