NAVJOT KAUR
(301404765)

ASSIGNMENT 2

Answer 1:- a. FALSE

Perfectly Rational is the ability to make good decisions and as can been seen in the example of vacuum agent when it does not know the state of adjacent square beforehand, i.e partial information but is still rational.

b. TRUE

Reflex agents selects the actions based on current percept and not on percept sequence. For example, in vacuum agent case, its decision is based on its current location and whether it contains dirt or not (from textbook). Hence, there is a task environment in which no pure reflex agent can behave rationally.

c. TRUE

Suppose an action where all actions bear same reward i.e, single state environment. So, there exist a task environment in which every agent is rational.

d. FALSE

Suppose a condition where infinite loops occur for any agent operating in partially observable environment. Infinite loops are unavoidable and can take up all the memory resulting in running out of memory.

e. TRUE

Suppose the equal reward case, where all actions in a deterministic task environment bear equal rewards, where agent is rational.

f. TRUE

It is indeed possible, take the case of an unbiased and biased coin (where result is always heads). If the player chooses heads for the unbiased case as well, then it is a rational behaviour in two distinct task environments.

g. FALSE

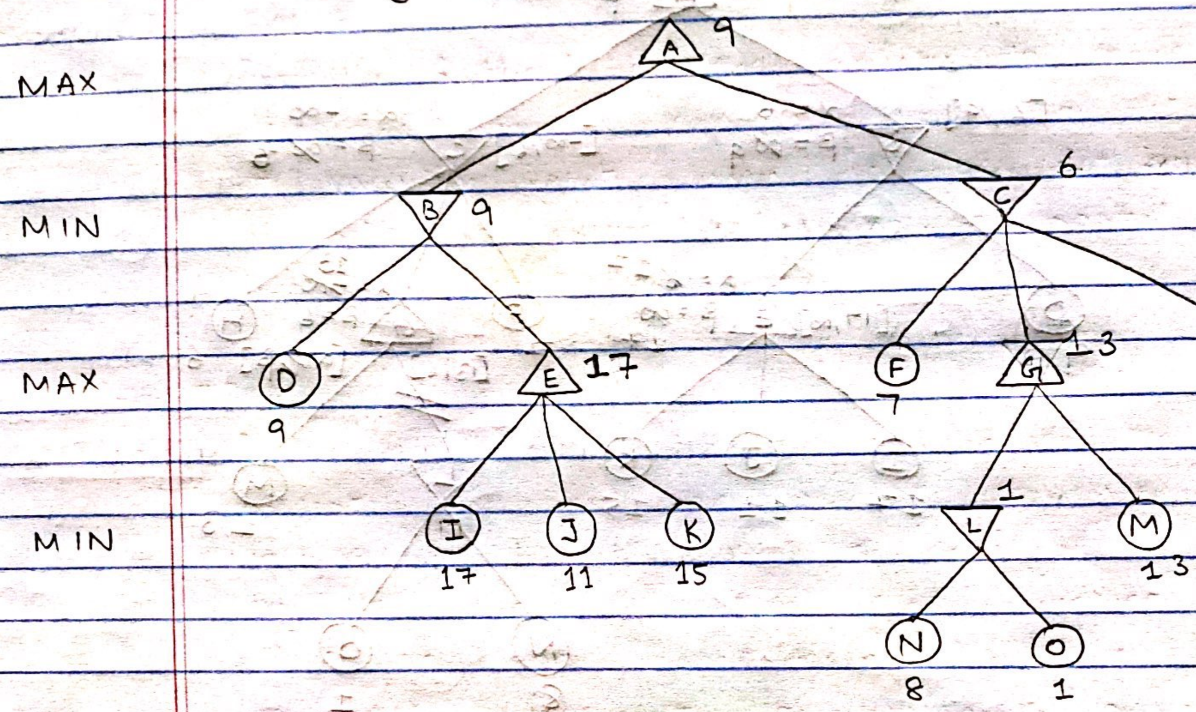
A vacuum agent that cleans and performs what it needs to is rational, however one that does not clean or do anything is not rational in an unobservable environment.

h. FALSE

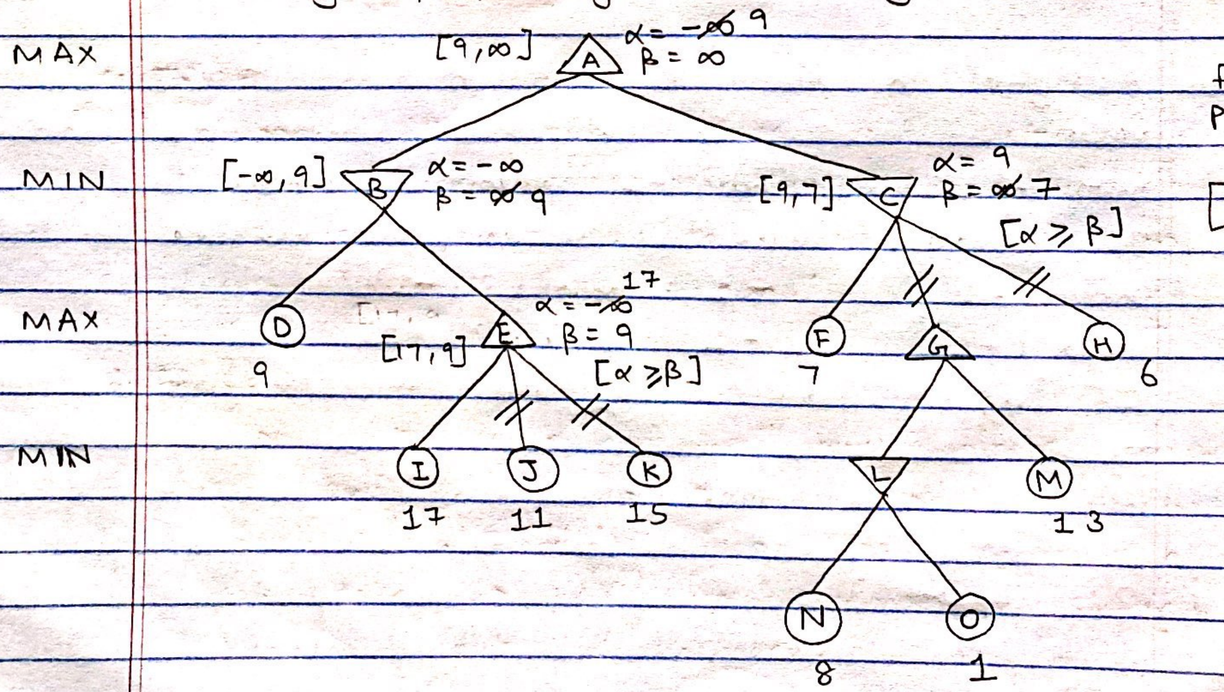
It is not true because a player can lose if the other player playing has better cards. More generally, both players are playing poker one has to lose, both can not win.

Answer 2 :-

a. Performing minimax algorithm, we have,

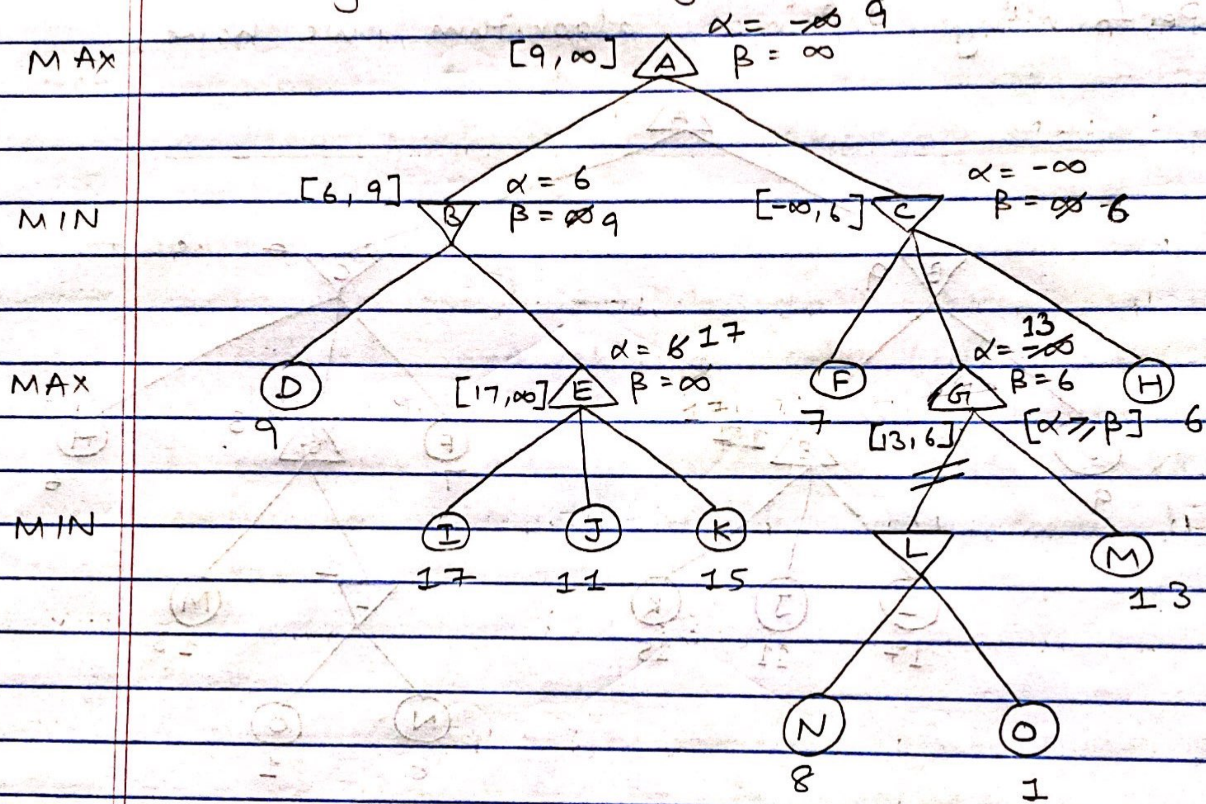


b. Performing α-β pruning [left to right]

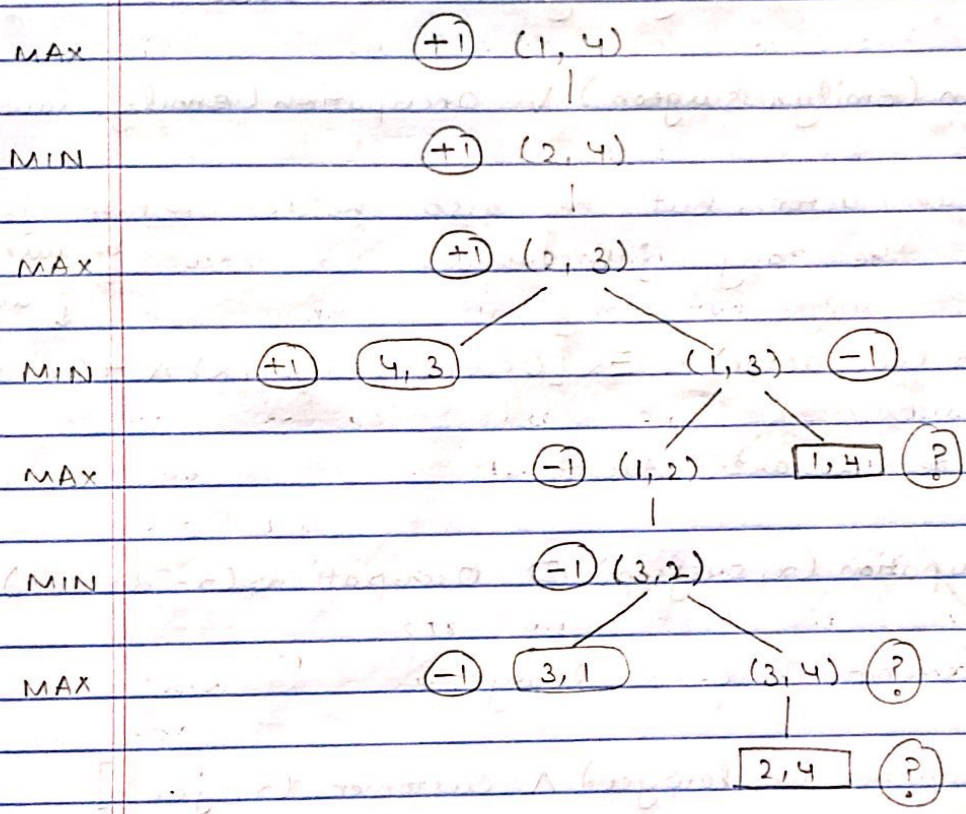


α = max
β = min
for pruning:
α ≥ β.
[symbol for pruning is //]

c. Performing α-β pruning [right-to-left] we have,



Answer 3 :- a. Complete tree :



Here ▭ symbolises loop state and ⬭ symbolises terminal state and the value in ○ denotes the game value.

(1, 4) means

A			B

A in 1st position and B in 4th position.

b. As can be seen in the above graph, the minimax value is denoted in a circle where +1 states that A wins and −1 states B wins.

The question mark (?) values are handled as follows:

If $\min(-1, ?) = -1 = \min(?, -1)$

If $\max(+1, ?) = +1 = \max(?, +1)$

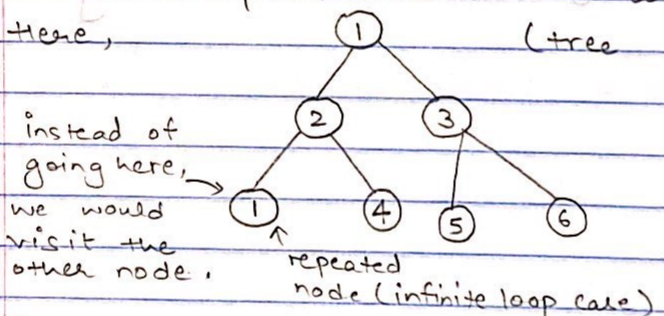
If ? is the value of successor, then the state will remain ?.

These values are as such because we assume if given a choice between ? and to win, a player will choose to win and avoid ? state.

c. ∘ As can be seen from the graph, the loop states will lead to infinite loops and due to this the standard minimax algorithm would fail because it uses depth-first search and would lead to infinite loops.

∘ It can be fixed by checking if the state that is being examined has occured before. If it indeed have occured before, then ? is used as the game value and is handled as discussed in part b. As a fix, there can be a structure such as map or an array that keep records of the states that are repeated so that we would check another state to avoid infinite loops.

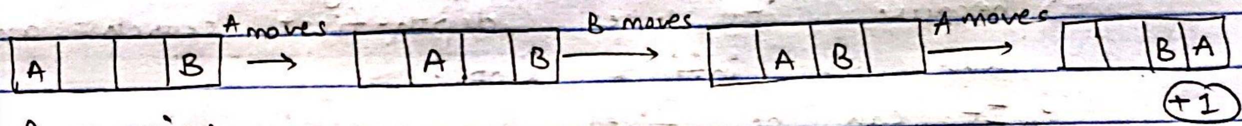
for example, take an array | 1 | 2 | 3 | 4 | --- |
Here,



(tree of the array)

∘ For all games with loops, the algorithm is not optimal, it can be for small cases however, not for larger games because keeping records of the repeated states will reduce the optimality. Also, there can be other cases where a game can have a draw apart from just winning and losing. So, it is not optimal in these games and other large games

d. In the 4-square game provided where A moves first, A has a winning strategy. It would like as follows:



A wins.
However, if B would have started first then B would win in the similar manner.

For general case, with n-square game, it can be deduced that (If player A moves first),
If n is even, then player A has a winning strategy and if n is odd, then player B has a winning strategy. for $n > 2$.

Answer 4 : a. Emily is a surgeon or a lawyer.
Using the vocabulary given, we have,

$\Rightarrow$ Occupation (emily, surgeon) $\vee$ Occupation (emily, lawyer)

b. Joe is an actor, but he also holds another job.
let x be any job, then,

because Joe is already an actor

$\Rightarrow$ Occupation (joe, actor) $\wedge$ $\exists x$ [Occupation (joe, x) $\wedge$ $\neg$(x = actor)]

c. All surgeons are doctors.

$\Rightarrow$ $\forall x$ [Occupation (x, surgeon) $\supset$ Occupation (x, doctor)]

d. Joe does not have a lawyer

$\Rightarrow$ $\neg \exists x$ [Occupation (x, lawyer) $\wedge$ customer (x, joe)]

e. Emily has a boss who is a lawyer.

$\Rightarrow$ $\exists x$ [Boss (x, emily) $\wedge$ Occupation (x, lawyer)]

f. There exists a lawyer all of whose customers are doctors.

$\Rightarrow$ $\exists x$ [Occupation (x, lawyer)] $\wedge$ $\forall y$ [Customer(y, x) $\supset$ Occupation(y, doctor)]

g. Every surgeon has a lawyer.

$\Rightarrow$ $\forall x$ [Occupation (x, surgeon)] $\supset$ $\exists y$ [Occupation(y, lawyer) $\wedge$ customer (x, y)]