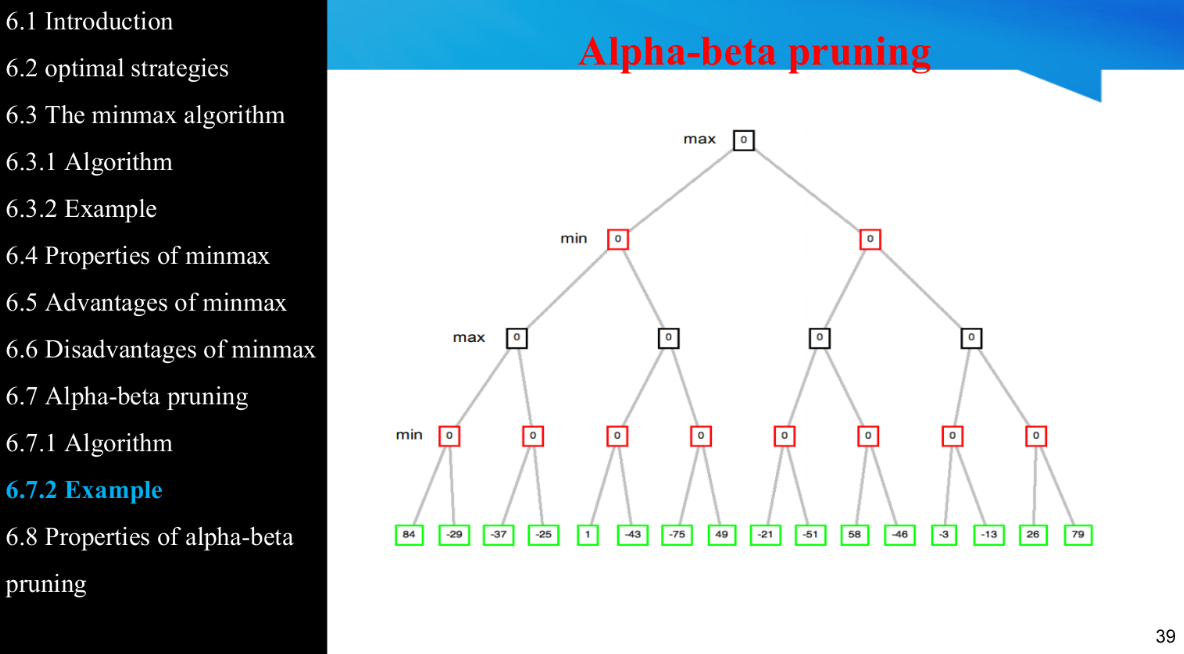
**Question Bank**

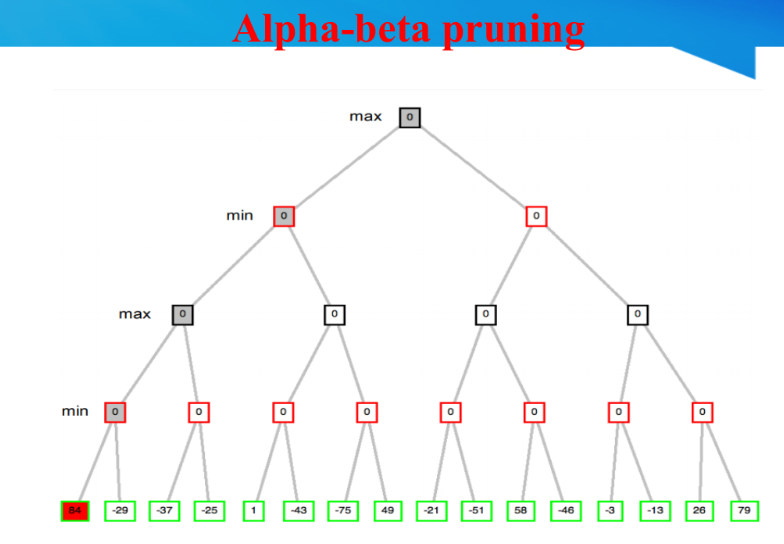
**Q1. Alpha beta pruning**

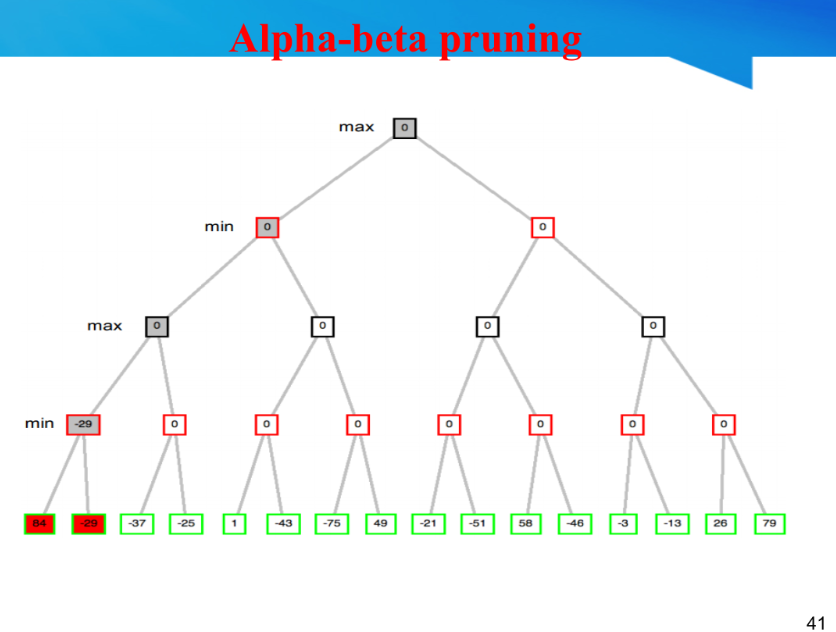
* Alpha is the best choice so far for player MAX. The highest possible value for this is demanded here.
* Beta proves to be the best choice so far for MIN, and it has to be the lowest possible value.

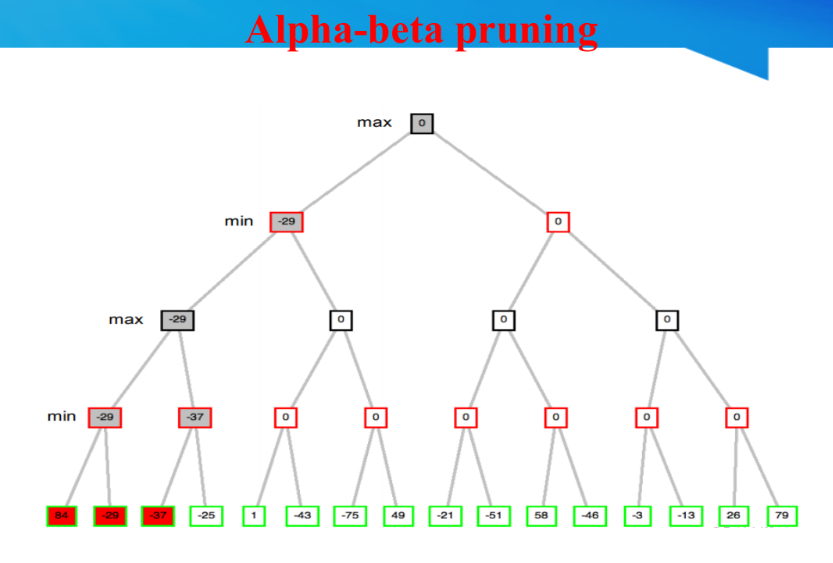
**ALGORITHM**

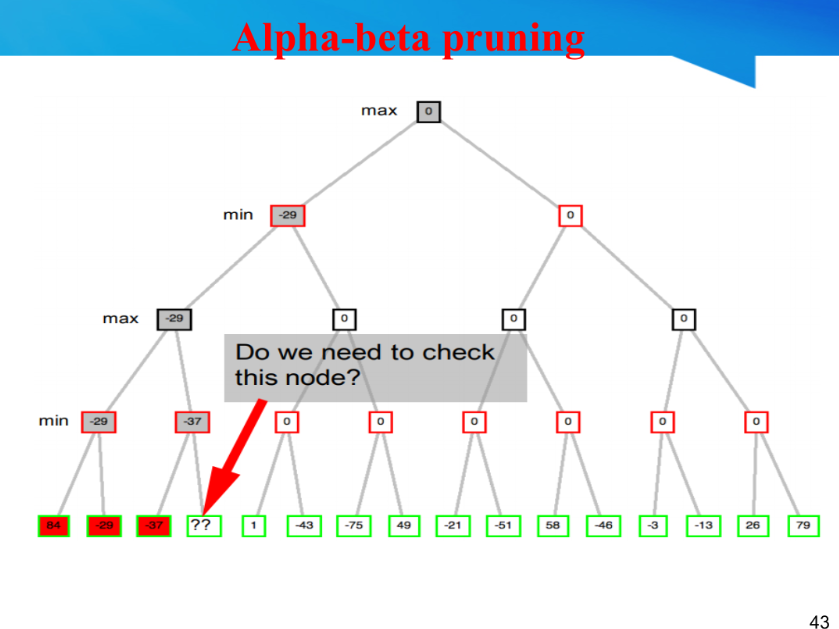
* Alpha is set to -infinity and beta to infinity.
* If the node is a leaf node, the value is returned.
* If the node is a min node, then for each children the minimax algorithm is applied with the alpha–beta pruning.
* If the value returned by a child is less, the beta sets beta to this value.
* If at any stage, beta is less than or equal to alpha do not examine any more children.
* If the node is a max node, the value of beta is returned. For each of the children apply the minimax algorithm with the alpha– beta pruning.
* If the value returned by a child is greater, the alpha set alpha to this value.
* If at any stage alpha is greater than or equal to beta, more children are not examined, and the value of alpha is returned.

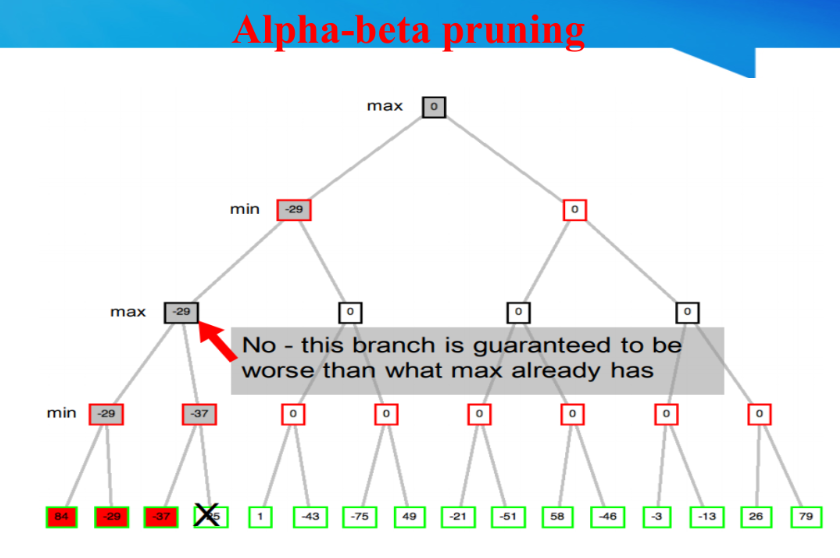
****

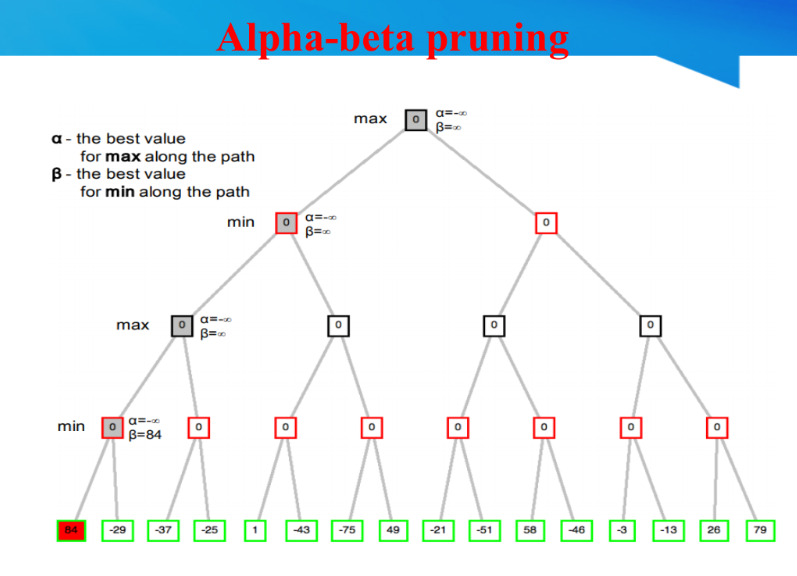
****

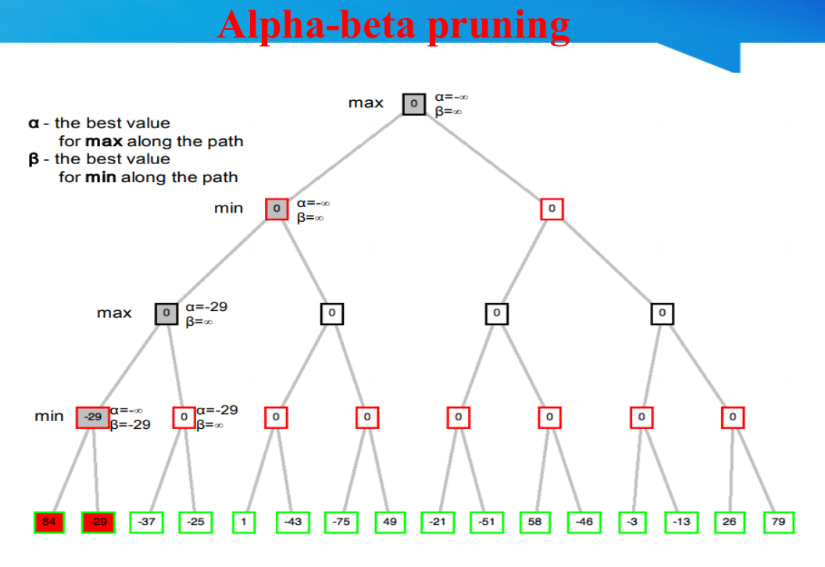
****

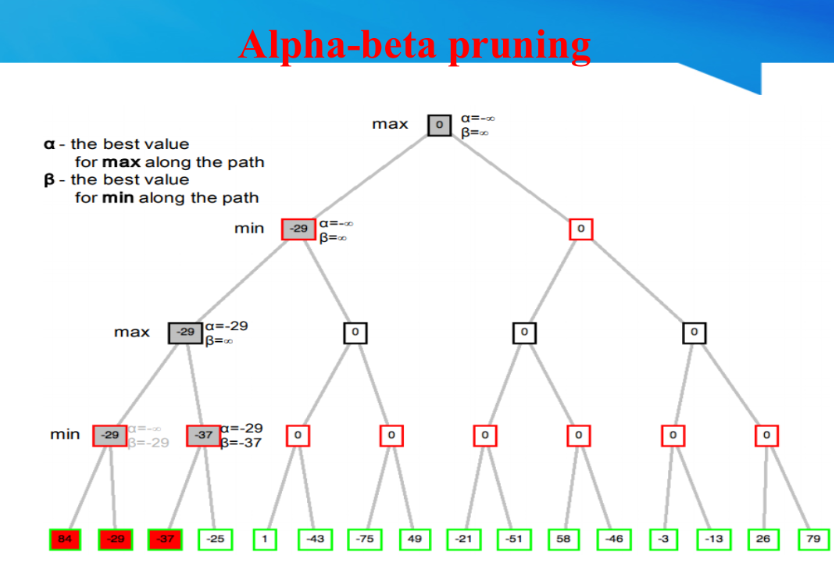
****

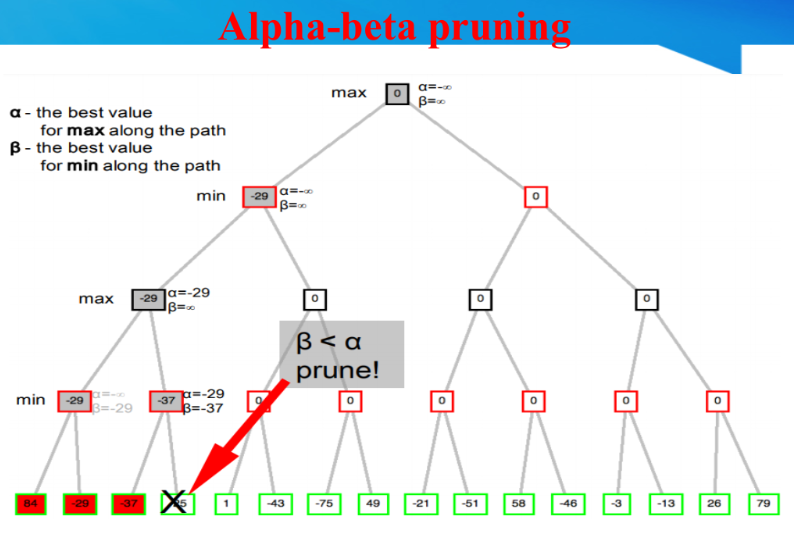
****

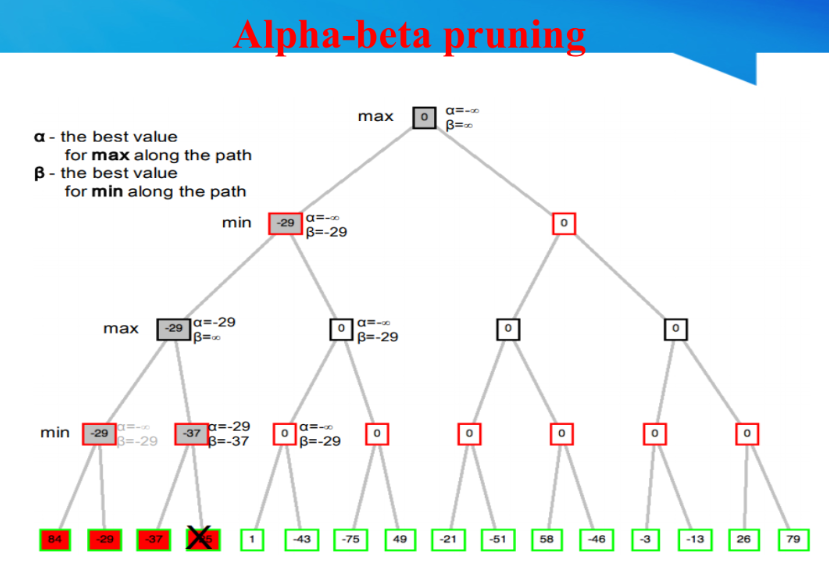
****

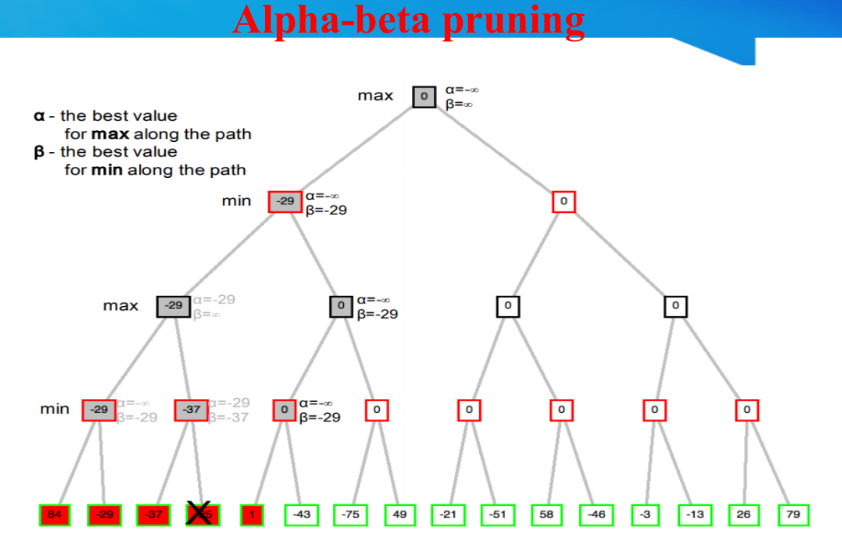
****

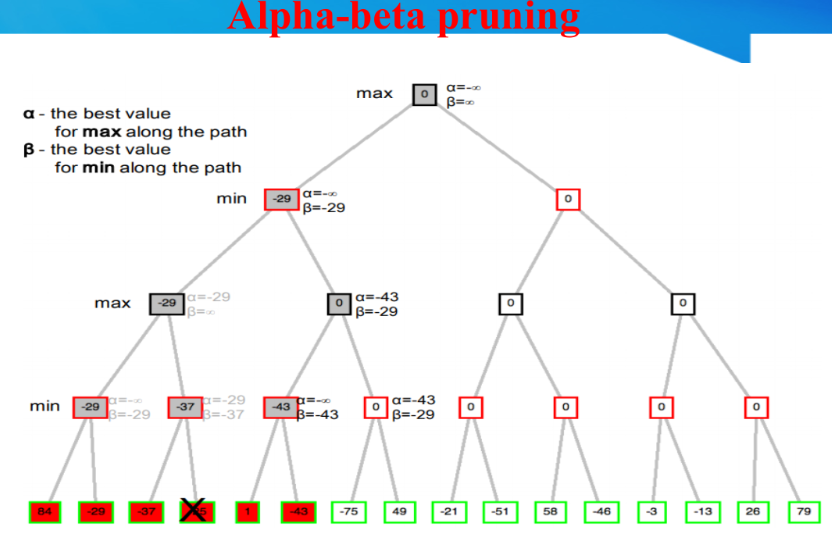
****

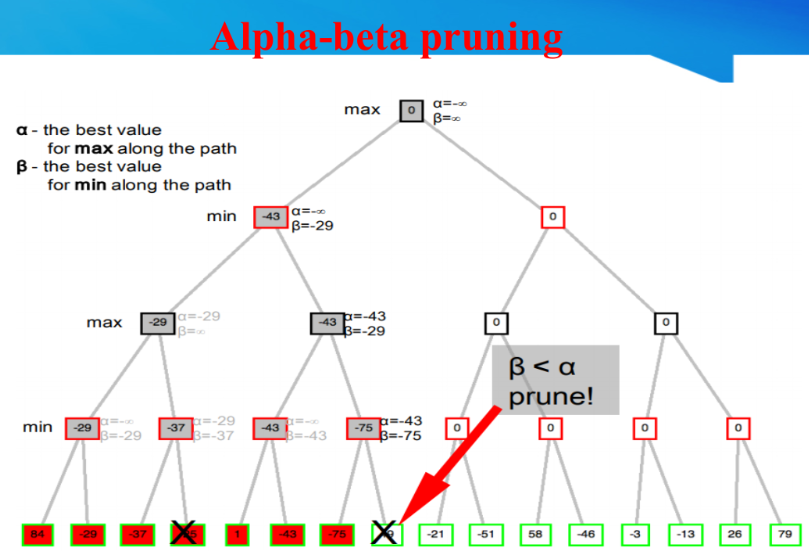
****

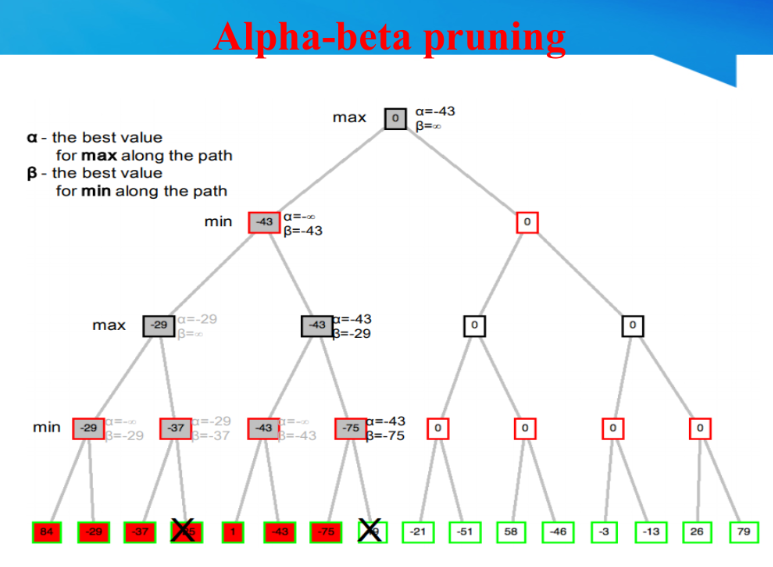
****

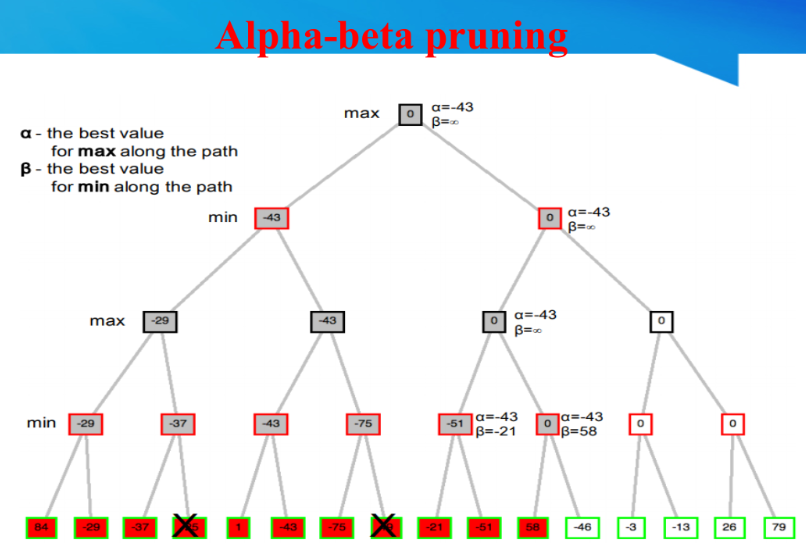
****

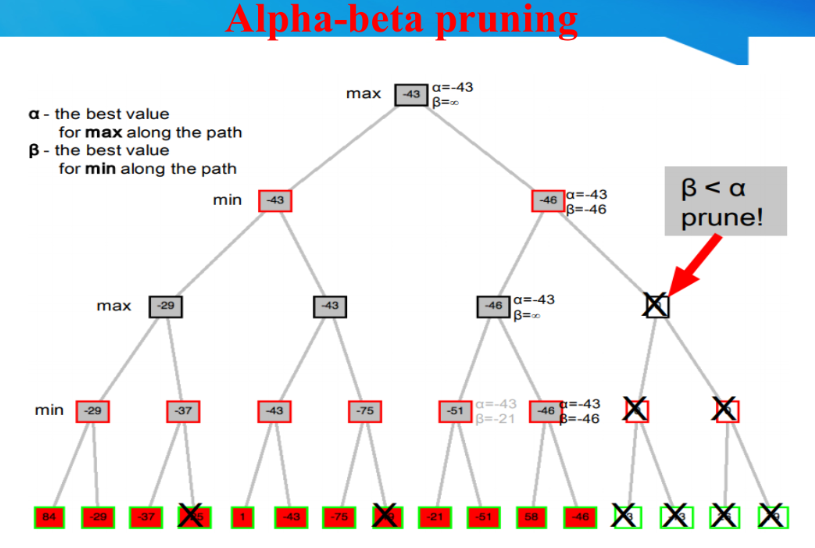
****

****

****

****

****

****

**For Alpha pruning 2nd eg - pg 56/70 and 43/87**

**Properties of Alpha-beta pruning**

1. Completeness: There is no completeness.
2. Optimality: This is against an optimal player. However, the fixed-depth version is not optimal.
3. Time complexity: With the perfect child ordering, O(bm/2), where b is the branching factor and m is the maximum depth or depth limit. Child ordering has a huge impact on the effectiveness of the pruning.
4. Space complexity: O(bm).
5. Pruning does not affect the final result (it is exact).
6. Good-move ordering improves effectiveness of the pruning (see
7. the last branch in examples).
8. Different orderings of the sequences of the moves may lead to the same state. Save the value of these “transpositions” to avoid the double work. With “perfect ordering,” time complexity = O(bm/2) doubles depth of the search.

**Q2. A star**

Pg 13-31/77 - <https://drive.google.com/file/d/1JkvT-4uCQuYZQMm263gJdP2xTbhMS2uH/view>

Pg 1-106/123 - <https://drive.google.com/file/d/1m0t-BEH-_-d6Aa7eRNxdfXSf8EmjqkQx/view>

Pg 47-95/104 -

<https://drive.google.com/file/d/14JxCtzTNdmEd8ugJ229urVPEtwe52af5/view>

**Q3. Cryptarithmetic**

**PPT-** [**https://drive.google.com/file/d/12DXLnDq57XQfLndhVooTkzW6al32TyIP/view**](https://drive.google.com/file/d/12DXLnDq57XQfLndhVooTkzW6al32TyIP/view)

**Q4. FopA Pg 43-70**

**PPT-** [**https://drive.google.com/file/d/1kDbMczi0\_1mGm\_YdLknHYuieewDB6GqZ/view**](https://drive.google.com/file/d/1kDbMczi0_1mGm_YdLknHYuieewDB6GqZ/view)

**Mod3 1-70 aayega imo pura hi padh lena :))**

**Q5. Difference between agents**

**ANS- Chapter\_2\_AI\_Agent.pptx pg18 onwards**

**For ref-** [**https://www.geeksforgeeks.org/agents-artificial-intelligence/**](https://www.geeksforgeeks.org/agents-artificial-intelligence/)

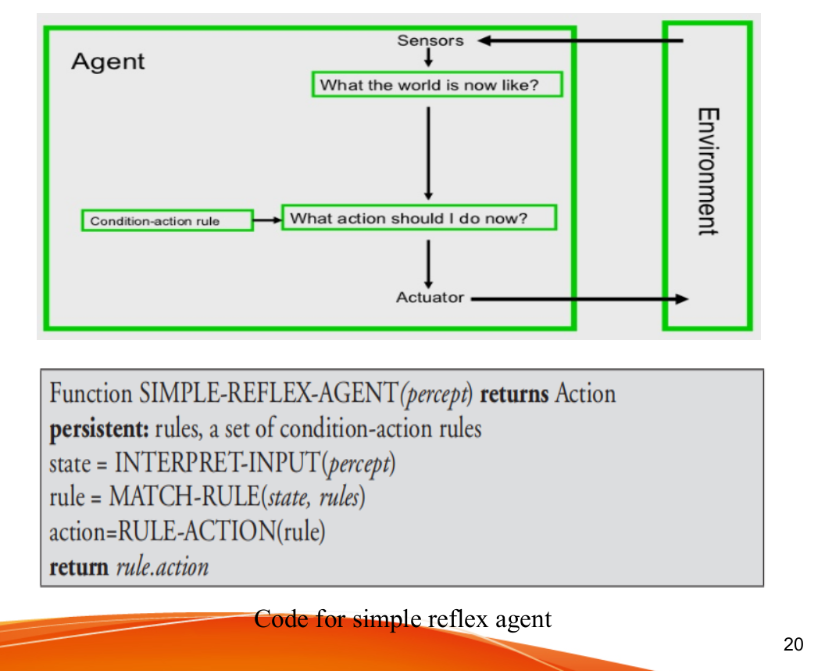
**Types of agents**

1. **Simple reflex agent**

* Simple reflex agent is said to be the simplest kind of agent.
* These agents select an action based on the current percept ignoring the rest of the percept history.
* These percept to action mapping which is known as condition-action rules (so-called situation–action rules, productions, or if–then rules) in the simple reflex agent.

It can be represented as follows: if {set of percepts} then {set of actions}

For example, if it is raining then put up umbrella



**Limitations**

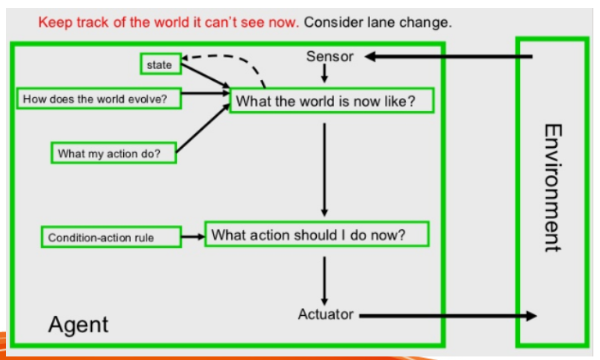
1. Intelligence level in these agents is very limited.
2. It works only in a fully observable environment.
3. It does not hold any knowledge or information of nonperceptual parts of state.
4. Because of the static knowledge based; it’s usually too big to generate and store.
5. If any change in the environment happens, the collection of the rules are required to be updated.

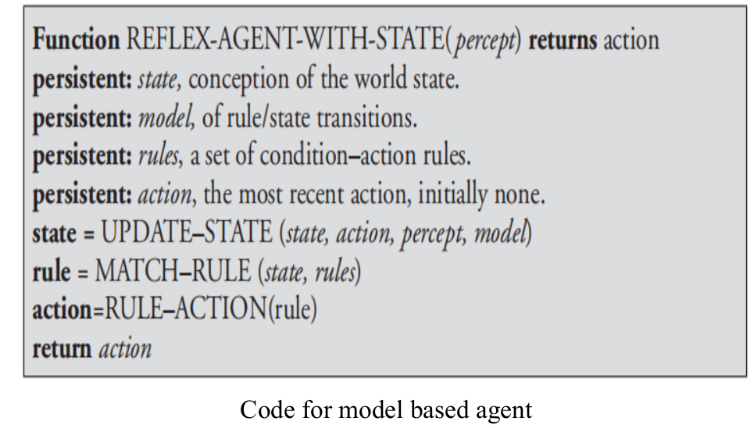
**2) Model based agent**

* Model-based agent is known as Reflex agents with an internal state.
* One problem with the simple reflex agents is that their activities

are dependent of the recent data provided by their sensors.

* On the off chance that a reflex agent could monitor its past states,and understand about the development of the world.

****

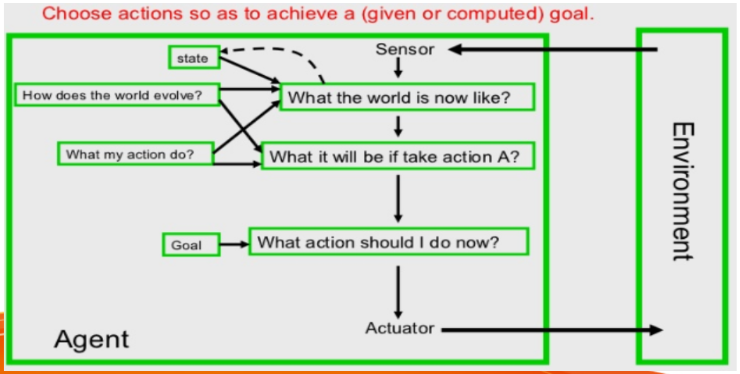
****

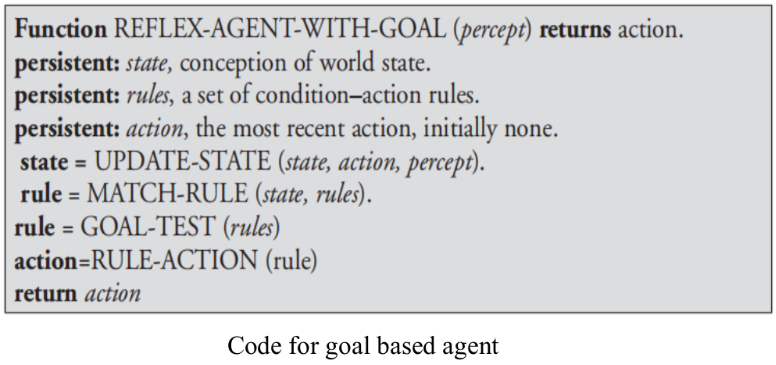
**3) Goal based agent**

* Indeed, even with the expanded information of the current

situation of the world given by an agent’s internal state,

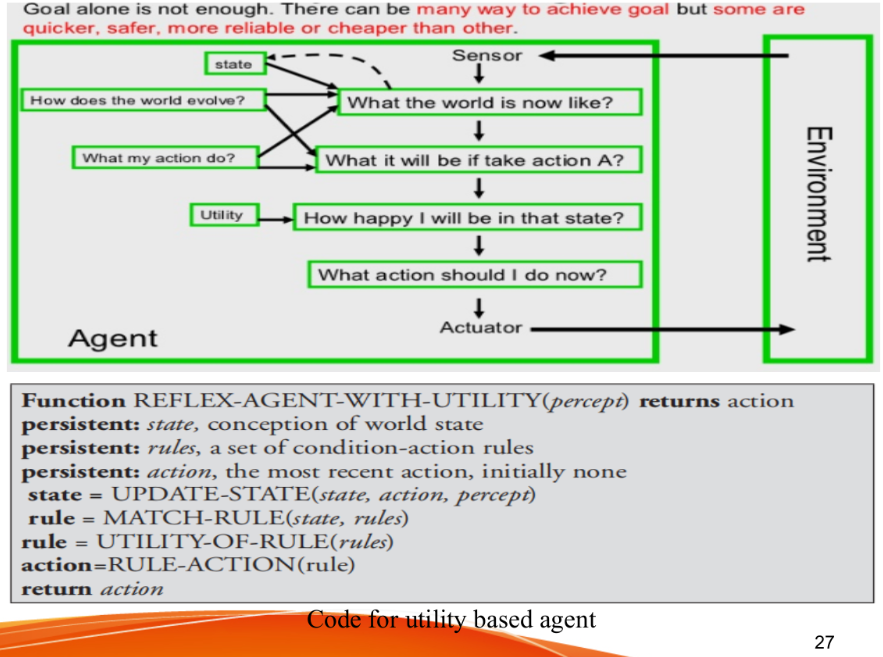
* The agent may not, in any case, have enough data to reveal to it.
* The proper action for the agent will regularly depend upon its
* goals.
* Thus, it must be provided with some goal information.

****

****

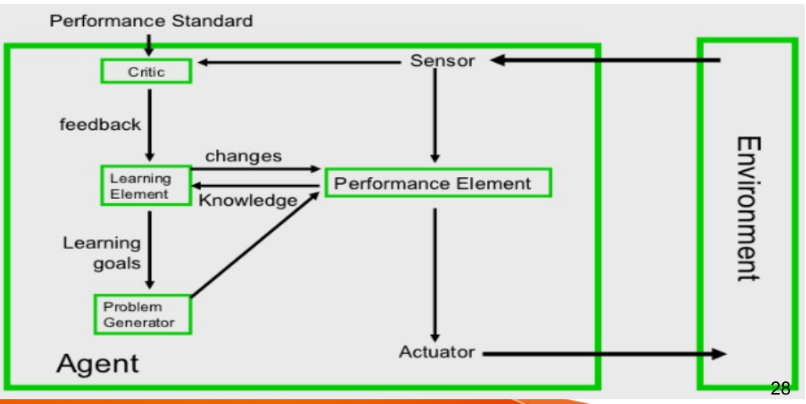
**4) Utility based agent**

* Goals individually are insufficient to produce top high-quality behavior.
* Frequently, there are numerous groupings of actions that can bring about a similar goal being accomplished.
* Given proper criteria, it might be conceivable to pick ‘best’ sequence of actions from a number that all result in the goal being achieved.
* Any utility-based agent can be depicted as having an utility capacity that maps a state, or grouping of states, on to a genuine number that speaks to its utility or convenience or usefulness.

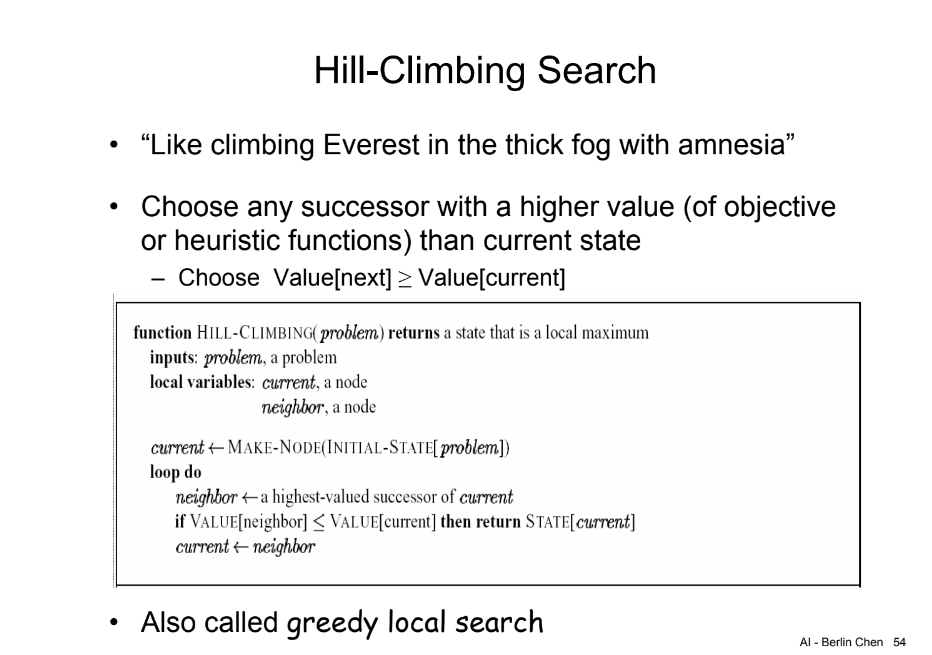


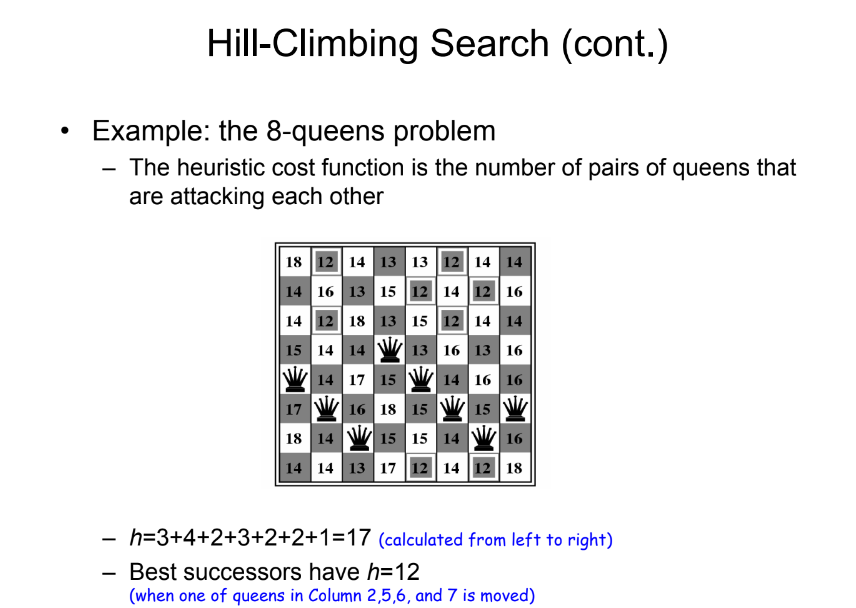
**5) Learning agent**

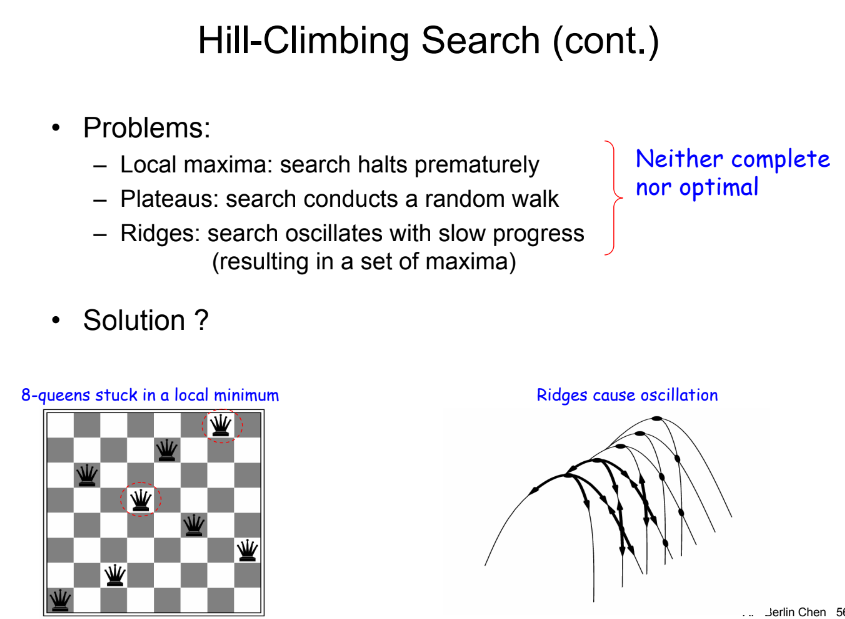
* By actively exploring and experimenting with their environment, the most powerful agents are able to learn.
* A learning agent can be further divided into the four conceptual components

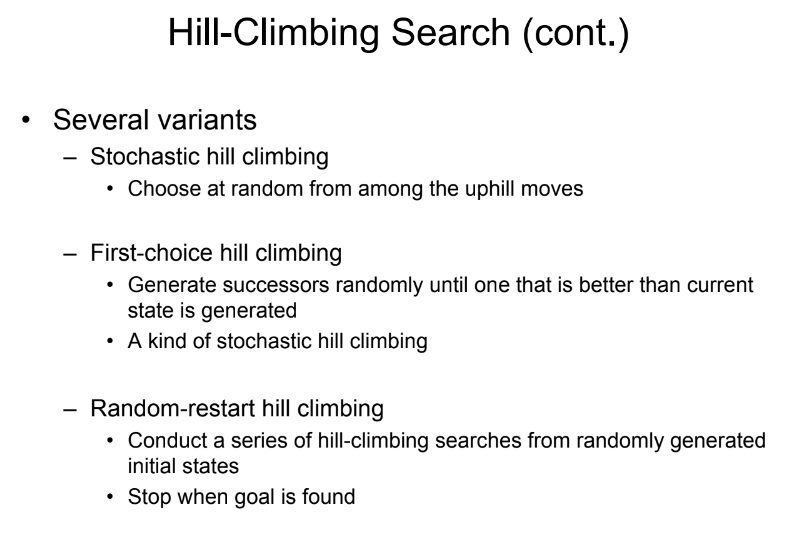
****

**Q6. Hill climbing limitations**

****

****

****

****

**Pg 24-39/104 -** <https://drive.google.com/file/d/14JxCtzTNdmEd8ugJ229urVPEtwe52af5/view>

**Q7. Water jug problem 3 water jug bacha hai dalna**

**Ref-**

[**https://www.geeksforgeeks.org/water-jug-problem-using-bfs/**](https://www.geeksforgeeks.org/water-jug-problem-using-bfs/)

[**https://www.geeksforgeeks.org/water-jug-problem-using-memoization/**](https://www.geeksforgeeks.org/water-jug-problem-using-memoization/)

[**https://www.geeksforgeeks.org/two-water-jug-puzzle/**](https://www.geeksforgeeks.org/two-water-jug-puzzle/)

**Q8. Inference rules with example pg21-34**

**PPT-** [**https://drive.google.com/file/d/1kDbMczi0\_1mGm\_YdLknHYuieewDB6GqZ/view**](https://drive.google.com/file/d/1kDbMczi0_1mGm_YdLknHYuieewDB6GqZ/view)

**Mod3 1-70 aayega imo pura hi padh lena :))**

**Q9. Time and Space Complexity of some algorithm**

**Q10. PEAS Description of soccer problem**

**PEAS**

Performance – which qualities it should have?

Environment – where it should act?

Actuators – how will it perform actions?

Sensors – how will it perceive environment?

**Robot soccer player (ppt slide 15/24)**

P: Winning game, goals for/against

E: Field, ball, own team, other team, own body

A: Devices (e.g., legs) for locomotion and kicking

S: Camera, touch sensors, accelerometers, orientation sensors, wheel/joint encoders

**Ques10**

**PEAS for Robot Soccer Player:**

Performance Measure (P): To Play, Make Goal & Win the Game.

Environment (E): Soccer, Team Members, Opponents, Referee, Audience and Soccer Field.

Actuators (A): Navigator, Legs of Robot, View Detector for Robot.

Sensors (S): Camera, Communicators and Orientation & Touch Sensors.

Environment:

Partially Observable, Stochastic, Sequential, Dynamic, Continuous and Multi Agent.

For ref-<https://www.geeksforgeeks.org/peas-description-of-task-environment/>