*Ch. 4: Informed search Algorithms:-

The difference between Uninformed and informed

- Uninformed search strategies find solutions by systematically generating new states and testing them against the goal.

- Informed search strategies use problem specific knowledge to find solutions in an efficient manner

---

*Best-first search:-

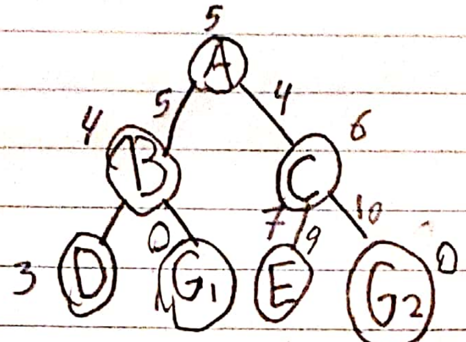
- It uses an evaluation function for each node

- The function is basically an estimate of "desirability"

- The fringe is a queue sorted in decreasing order of desirability.

⇒

★ Best-first search continued:



"We can see that there is a cost on the node and a cost on the path"

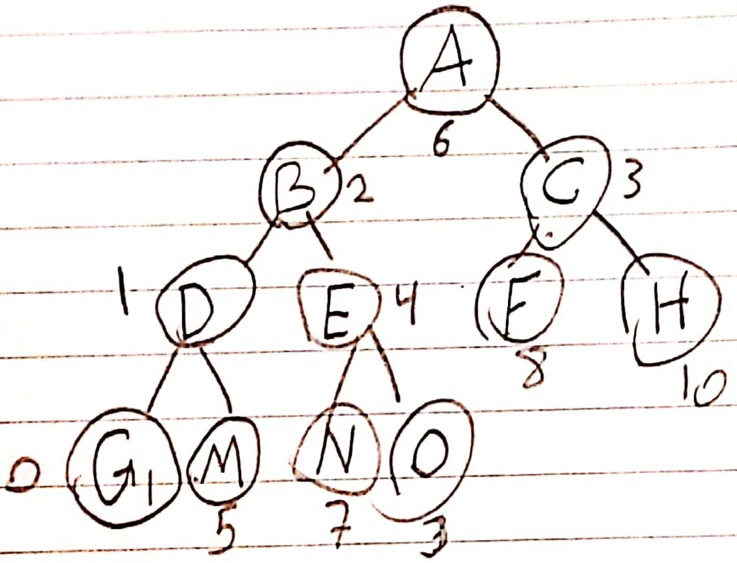
- When we perform this search, we look at the node cost then we return the path cost to the goal.

- We move from A→B, because B has less cost than C. then from B to G_1 and return the path.

* Greedy search:-

- Uses a heuristic function to estimate the cost from a node to the goal.

- Greedy search expands the nodes that appear closest to the goal.



- Similarly to Best-first search, we use the heuristic and explore the cheapest one first. So, the search here will return:

A → B → D → G_1

A search :-

- It should avoid expanding paths that are already expensive.

- It's very similar to UCS, but with the difference of the heuristic

- The heuristic should never overestimate the cost.

- A* score = cost of path + heuristic

- Whenever we expand, we look at the A* score and expand the cheapest we do that for the whole tree and chose the lowest A* score to the goal node.

- A* is an optimal search algorithm.

* Relaxed Problems:-

- A way to derive the heuristic from the exact solution cost of a relaxed version of the problem.

* Local search algorithms:-

* Hill climbing:-

- Hill climbing is a heuristic search used for mathematical optimization problems in AI

- It tries to find a good solution, however, the solution <u>may not be the optimal global maximum</u>

* Simulated annealing:-

- It's an effective and general form of optimization. The algorithm is basically hill climbing except instead of picking the best move, it picks a random move.

- A parameter T(temperature) is used to determine probability. T starts high then decreases; Less values of T indicates a higher hill

☆ Local beam search:-

- Keep K states instead of 1; choose top K of all their successors

- If any one of the successors is a goal, the algorithm halts.

- The K states are generated randomly.

☆ Genetic Algorithms:-

- They basically are stochastic local beam search + generate successor from pairs of states.

eg,

sol. here is BC

A	B	C

sol. here is DE

D	E	F	G

genetic algorithm result

B	C	D	E