```
A_Star(start, goal):
1. open_set = {start}
2. Came_From = empty map
3. For each node n:
      $g(n) = \infty$
      $f(n) = \infty$
4. g(start) = 0
5. f(start) = g(start) + h(start)
6. while open_set is not empty:
      a. Current = node in open_set with lowest f scor
      b. If Current == goal:
            Return Reconstruct_Path(Came_From, Curr
      c. Remove Current from open_set
      d. for each of Current:
         i. Tentativ_g = g(current) + d(current, negibo
         ii. if Tentativ_g < g(neighbor).
               Came_From[negiher] = current
               g(neighbor) = Tetative_g
               f(neighbor) = g(neighbor) + h(neighbor)
               If neighbor not in open_set:
                  Add neighbor to open_set
7. Return failure (no path found)
```

Tabu Search

```
1. Choose an initial Soulution S

2. Set Best_Soulation = S

3. Initial Tabu-List as empty

4. Repeat until Stopping Condition met.

    a. Gen Neighborhood(S)
    b. From neighbors, Select best candidate S'
                                        not in List
    c. if Cost(S') < Cost(Best_Soulution):
           Best_Soulotion == S'

    d. Ubdate Tabu_List with move from S to S'

    e. If Tabu_List exceeds sizeLimit:

           RM oldest entry
    f. Set S=S'

5. Return Best_Soulution
```

Q1: what is the difference between bagging and boosting?

Answer:

Bagging: Reduces variance by training models in parallel (eg, Random forest)

Boosting: Reduces bias by training models sequentialy, each focusing on correcting the errors of the privious one. (eg XGaBoost).

Q2: Explain the concept of overfitting and how to prevent it.

Answer:

overfitting: occurs when a model learns the traing data too well, including noise and outliers, and fails to generlize to new data.
preventation methods include cross validation, prunning, dropout, regularization, and more data collecting