1.) Uninformed Search

   a. Know what a search problem is and how to formulate one (i.e., state space, possible actions, transition model, goal test, and path cost)

   b. Know the characteristics of, and differences between, Breadth-First Search (BFS), Depth-First Search (DFS), Uniform-Cost Search (UCS), and Iterative-Deepening Search (IDS).

   c. Know the respective complexities of these respective algorithms.

   d. Be able to execute these searches on example problems.

2.) Informed Search

   a. Know what a heuristic is and how they aid searching.

   b. Know the characteristics of, and differences between, Greedy Best-First Search, A Search, and A* Search.

   c. Know the respective complexities of these respective algorithms.

   d. Be able to execute these searches on example problems.

   e. Understand what makes functions admissible and consistent.

   f. Understand local searching and hill climbing.

   g. Understand simulated annealing, how it helps with local optima, and be able to execute it on examples.

   h. Understand genetic algorithms, how it uses fitness to determine parents used in mating. Know how it differs from other local searches, and be able to work through example problems. Also, be aware of various selection techniques.

3.) Game Playing

   a. Be familiar with the types of games (e.g., zero-sum, discrete, finite, deterministic, and perfect information).

   b. Understand what a utility function is and how it’s used in the greedy search.
c. Understand the **Minimax** algorithm and how to execute it. Also know the complexity of it.

d. Be able to work through examples using Minimax.

e. Know what a **static board evaluation (SBE)** is and how/why it’s used.

f. Know what **alpha-beta pruning** is and why it’s used.

g. Be able to execute alpha-beta pruning on examples.

h. Understand how limited time comes into play with game play algorithms and how it’s handled. For example, understand what the **horizon effect** is and how to deal with it.

i. Have an intuitive understanding of how game play algorithms can handle randomness.

4.) **Unsupervised Machine Learning**

a. Understand what learning is and what machine learning aims to achieve.

b. Understand what **inductive learning** aims to achieve.

c. Understand what the following pieces of inductive learning are and how they are used in the learning task:

   i. \( x \) (an instance)

   ii. \( y \) (the label)

   iii. **hypothesis function** and how it relates to the unknown function \( f \)

   iv. **feature vector**

   v. **feature/attribute**

   vi. **feature space** (\( D \) dimensions)

   vii. **training sample**

   viii. **test sample**

d. Understand how sampling can affect learning (i.e., why we need **iid** sampling).

e. Know the difference between **supervised learning** and **unsupervised learning**.
f. Understand how clustering is an unsupervised task.

g. Know about Hierarchical Agglomerative Clustering (HAC) and be able to use it with single-linkage, complete-linkage, and average-linkage. Understand and be able to draw the associated binary tree.

h. Know the K-Means Clustering algorithm and be able to use it. Know what a centroid is and how it pertains to the algorithm.

i. Know what Mean Shift Clustering is and how it works.

j. Understand what distortion is and how it can be used to determine how good a particular clustering is.

5.) Supervised Machine Learning

a. Understand how supervised machine learning is different from unsupervised.

b. Understand what the following ideas are and how they relate to unsupervised learning:

   i. Label
      1. Positive example
      2. Negative example
   
   ii. Classification
   
   iii. Regression
   
   iv. Classes

c. Understand what kNN is and how to use it for classification and regression.

d. Know what inductive bias is.

e. Understand what SVMs are. In particular, know what the following mean:

   i. Margin

   ii. Maximizing the margin

   iii. Support vectors

   iv. Linear separability
v. Slack variables

vi. Trade-off parameter (aka C-value)

vii. Kernels

1. Why we use them

2. How they relate to the dot product encountered in the second formulation