

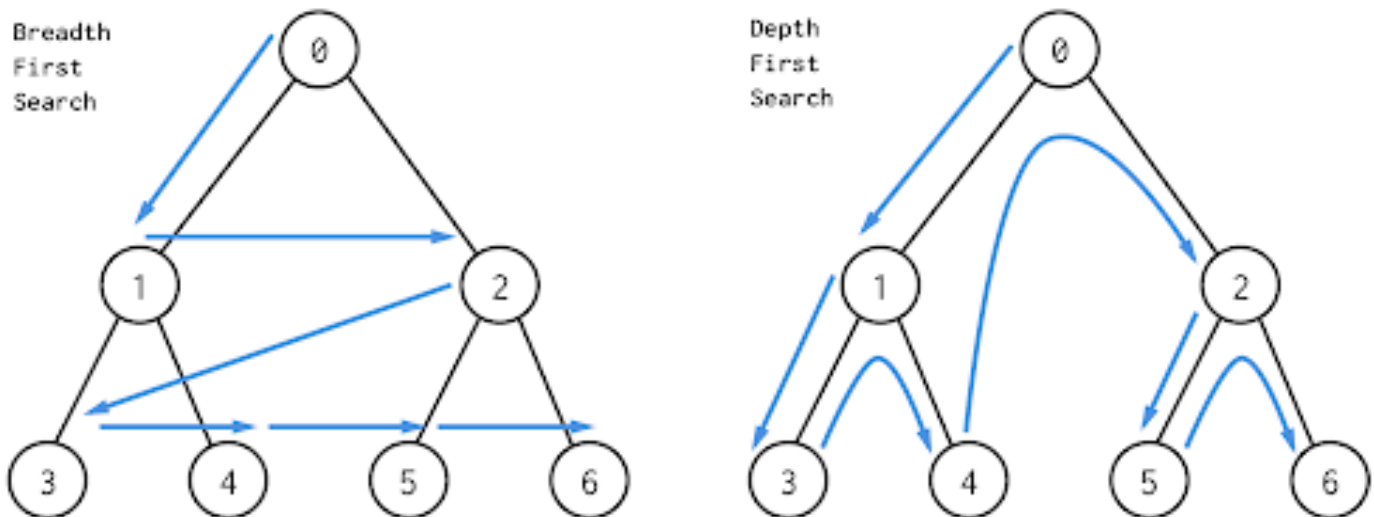
Topic: AI Search [Subsection of Unit (2-3 days)]

Hunter CS Advanced Certificate - Topics

Sam Keener & Jack McCoy

BFS vs. DFS

Uninformed Searches (also known as blind searches) - They are ignorant of any other variables, only the nodes they are presented with.



Imagine that you are looking for a treasure in a cave. It is in one of the rooms, but you don't know which one. Your first decision is to choose one of three paths. After you choose a path and reach another room, you are presented with another fork in the path. Do you continue forward, or go back and check the other options?

Breadth First Search - this algorithm constantly backtracks to find the shortest possible path. Will check each option connected to the starting node. If none match the goal, it will look at each node in turn, checking all of its available options for the solution before moving to the next step. The goal is to find the **SHORTEST POSSIBLE PATH** to the solution, but may take longer than depth first search. The information is stored in a queue data structure.

Depth First Search - this algorithm will always keep moving left and down until it can't any more starting on the left-most edge. Then it will backtrack as little as it possibly can before exploring another branch all the way to the end. It will eventually check every node, and will find the treasure if it exists, but not necessarily the most optimal route. The information is stored in a stack data structure, pushing the current node on and after popping the previous node off. In general, it finds the solution quicker, but not always on the shortest path.

Both will find a solution if it exists, but BFS will always find the shortest (optimal) path, and DFS will not always find the shortest (not optimal), though occasionally it will find it anyway.

A*

Informed Searches - Has a method for evaluating the closeness of a given node to the solution, thus making an informed decision about which way to go.

A* search - (A star) Each node has a value that represents how “close” it is to the target node. This is called an **h-value**, which stands for heuristic, and is an estimate calculated by some formula that accounts for some chosen variables. What that formula is obviously depends on the situation, but it is extremely important that the h-value of a node is NOT GREATER than the ACTUAL cost to travel from that node to the solution. The path between two connected nodes also has a **g-value**, called a “cost”, basically what it takes (distance, computing time, etc.) to go from one node to another.

The A* search algorithm uses both of these values to calculate the best option when traveling between two nodes. It looks at each option, adds the “cost” of traveling to that node to the heuristic (how close it is to the target) of that node, and whichever option has the lowest sum gets chosen! It always explores branches

Metaphors: finding cheapest air travel from one city to another, identifying the shortest path on a rail system, exploring the fictional cave system for treasure

Differentiation – (limited in lesson plan, can tailor to your unique student’s needs. Recommendations employ guided notes, Skeleton Code, and student handouts.

Student Resources:

Search Algorithms in AI

<https://www.geeksforgeeks.org/search-algorithms-in-ai/>

Difference between BFS & DFS

<https://www.youtube.com/watch?v=VXMsgBLzz68>

A* search (longer video)

<https://www.youtube.com/watch?v=ySN5Wnu88nE>

A* search (shorter video)

<https://www.youtube.com/watch?v=vP5TkF0xJgI>

Lesson Plan Day 1

Topic: AI Searching: Uninformed Searches (BFS and DFS)
Objectives: SWBAT emulate a BFS and DFS on a given data set and compare/contrast BFS and DFS.
EQs: What is Artificial Intelligence? How do computers make decisions?
Key Vocabulary goal, optimal
Agenda: *Do Now/Opening/Bellringer/whatever: What is artificial intelligence? *Intro & Hook: Cave exploration *Unplugged activity - tracing map *Coding exercise - coding DFS/subgoal for BFS *Closing - compare/contrast
Opening - 5 minutes
Students will be asked to take out a KtS and write a 1-2 sentence answer to the Essential Question: "What is artificial intelligence?" Students will share their answers with their partner, then volunteers (or volunTOLDs) will be asked to share out. Utilize whatever discussion protocols you wish and cut it short if you're pressed for time.
Intro & Hook: Cave Exploration - 15 minutes
<p>Students are hopefully familiar with the concept of looking for a treasure in a cave. Present slide 6 (a cave room with 3 connections) and ask students which path they would choose and why. <i>A human might choose a path for specific reasons (stairs, light, look, etc.) or arbitrary reasons.</i> Ask students which path they think a computer would choose and why. <i>Students may argue that a computer would probably choose path A first because it is first alphabetically, or they may say that the computer would choose arbitrarily.</i> Either way, the students can then vote on which path they'd like to choose. Regardless, flip to slide 7, showing the room with two connections. At this point, students have the choice to either continue down one path, or go back and check the other paths. Highlight these two options, as well as any other strategies for exploring the caves that students come up with. Undoubtedly, at least one student will agree that going as far down a path as possible before turning back is ideal. Using the map of the cave on slide 11, demonstrate how the DFS works by asking students to predict which node will be visited next on a map of the cave in the slides (i.e., 2nd level left most, or to the left/right). Optional extensions: Relate it to chess (tie it back to AI) or relate it to a binary tree traversal (pre-order traversing) Reveal that this type of search algorithm is called a DEPTH FIRST SEARCH. Most likely a student will have suggested doubling back and checking the other options before going further down the path. Introduce the rules for the Breadth First Search. Show slide 14 to show the rules of the BFS and ask students to predict which node the BFS will check next as it works its way through the algorithm. After traversing the entire cave, move on to the unplugged activity with the different structure (secret tunnel)</p>

Unplugged Activity: Exploring the cave with DFS and BFS and Optimal Path - 15 minutes

Students will be given a handout and will be asked to use the DFS algorithm for exploring the cave (this time with a secret tunnel that connects back to a previous node), by drawing a path that shows the order that each node is visited. They will write the numbers inside the nodes based on where they are in the order (with the entrance room being number 1). Students may differ on where they think the path will lead them, which can lead to insights on why certain nodes are chosen (why left to right) and students can reach their own conclusions.

Students will answer questions about best and worst case scenarios for each type of search. They will reflect on which method will find the SHORTEST route to the treasure. Will it always find the shortest route? What if there are multiple goal nodes? Which one is better? Which one is faster? Which one might require more computing time?

Practice: Coding Exercise - 10 minutes

Students will be asked to work with their partner to create subgoal labels for the Depth First Search and the Breadth First Search on their KtS. Depending on where you are in the curriculum and what students have already programmed, you can give them a partially completed program to attempt to actually code DFS or BFS as extensions or homework assignments.

Lesson Plan Day 2

Topic: AI Searching: Informed Searches (A*)
Objectives: SWBAT emulate an A* search on a given data set and compare/contrast informed vs uninformed search.
EQs: What is Artificial Intelligence? How do computers make decisions?
Key Vocabulary informed/uninformed, cost, heuristic, A*
Agenda: <ul style="list-style-type: none">*Do Now/Opening/Bellringer/whatever: Compare/Contrast DFS vs. BFS*Intro & Hook: Cave exploration*Unplugged activity - tracing map*Coding exercise - coding DFS/subgoal for BFS*Closing - compare/contrast
Opening - 5 minutes
Write one benefit and one limitation of both BFS and DFS from yesterday's lesson.
Intro/Hook - Cave of Monsters - 5 minutes
<p>Reintroduce the cave from yesterday.</p> <p>"What if we had some information about the location of the treasure? Imagine you have a radiation detector and you know that the treasure is radioactive (you're wearing a hazmat suit so you're all good). Would that help you choose which way to go?"</p> <p>Students would correctly surmise that they could use the radiation detector to follow the right path. This is called an "Informed Search", where the computer has some information regarding the location of the goal node. BFS and DFS searches operated under the assumption that we had NO information about our goal node, what do you think those types of searches are called?</p> <p>"Now imagine that you get to a fork in the tunnel and you see THIS: Show slide 25.</p> <p>You're getting strong radiation readings from the direction the dragon is protecting, very faint readings from the kitten path, and some moderate readings from the path with the obvious booby trap. Which way would you choose?"</p> <p>Students may give different answers depending on their particular bravery.</p> <p>Is there a risk/reward structure here? Could you justify choosing each path?</p> <p>"Let's put some values on the closeness of the path to the treasure and the inherent cost/risk of taking that path. h represents how close the path is to the treasure, the lower the better, and g represents the cost of traveling to that node. Obviously we can see that the dragon represents the highest cost, but it will bring you very close to the treasure. The kittens offer practically no cost, but won't really bring you closer to the treasure. The death pit seems easy enough to avoid as long as you tread carefully, but won't bring you as close to the treasure as following the path with the dragon. Which path should you choose?"</p>
A* search - 5 minutes
The A* Search is a search method that utilizes this information to make an informed decision about what is likely the best path to take. It involves two values, the g-value (the cost of traveling from one node to

another) and the h-value (an estimate of the total cost to travel from that node to the goal node). The A* search algorithm will always choose the path that it believes will get us to the goal at the lowest cost. To figure out what that is, it will add the cost of traveling to each node to the heuristic (the minimum total cost from that node to the goal) at each respective node. Whichever path has the lowest sum will be chosen, but the sums of each of the other paths will be remembered. It will expand the options of the chosen node and continue to utilize the cost+heuristic formula to travel to the next node until the total cost of the proposed journey exceeds the minimum total cost of one of the unexpanded options, at which point it will go back and expand that option until the total cost is greater. If it reaches the goal and the total cost is lower than all of the minimum total cost of all the remaining unexpanded paths, then it will return that as the shortest optimal path.

For this reason, it is absolutely imperative that the heuristic is less than or equal to the actual total cost of traveling from that node to the goal. Otherwise, the A* algorithm simply will not always work.

Railroad Demonstration - 10 minutes

Teacher-guided demonstration of a real-world application of A* search using railroads in Romania.

Focus on the decision making process. It would help to keep track of the total cost of our “current” journey with a variable written out either on the board or have students keep track.

Slide 30. Explore Zerind, Sibiu and Timisoara. Add g and h values to get the f-sums. Choose Sibiu as it has the lowest f sum. Even though Sibiu has the highest g-value, the sum of the h and g values is the lowest.

Slide 31. From Sibiu, expand to Arad (backwards), Oradea, Fagaras, and Rimnicu Vilcea. Note that the g-value includes the TOTAL COST from the starting node.

Slide 32. Continuing from Rimnicu, the lowest sum is 417, which is higher than the f-value of Arad-Sibiu-Fagaras, so we will choose to expand THAT path instead.

Slide 33. From Fagaras, we can travel to Bucharest, but our total cost has an actual value of 450, which is HIGHER than the potential minimum value had we chosen to expand pitesti, so we are going to continue with our work.

Slide 34. Upon expanding Pitesti, we see a path from the Starting node to the goal-node that has a lower overall cost than the previously discovered path. Since this cost is also lower than the MINIMUM costs we could hope to achieve by exploring any other previously unexplored path, we KNOW that we have the OPTIMAL route!

Practice: 30 minutes

Students will be given another map of the cave handout, this time with g and h values for each path and node, and will be asked to emulate an A* search on the map. There may be some debate about what to do on the second step, whether to continue from node 7 to node 10 or to go back to node 2, as there is a tie for total value vs. minimum total value. Students can come to their own conclusions about which path to follow (come up with a tiebreaker). An example tiebreaker could be fewer nodes traveled so far.

After they are finished with the map and a partner discussion, students will work with their partners to subgoal label the A* search as if they were going to code it.