

(b)

States: The set of actors guessed so far and the last actor guessed, i.e. the pair $\langle X', x \rangle$, where $X' \subseteq X$ is the set of guesses, and x is the last actor.

Initial state: There have been no guesses, and the first player may begin by guessing any actor, i.e. $\{\langle \Phi, x_i \rangle \mid x_i \in X\}$.

Successor function: Add an actor y to the current set of guesses such that y is a co-star of x in some movie, i.e. $\text{successor}(X, x) = \{\langle \text{add}(y), (X \cup \{y\}, y) \rangle \mid y \text{ co-stars with } x\}$.

Goal test: Any state with no successor state. That is $\langle X', x \rangle$ such that there are no actors y co-starring with x that have not already been guessed.

Cost function: The number of guesses so far, $|X'|$.

2 Blind search (50 pts)

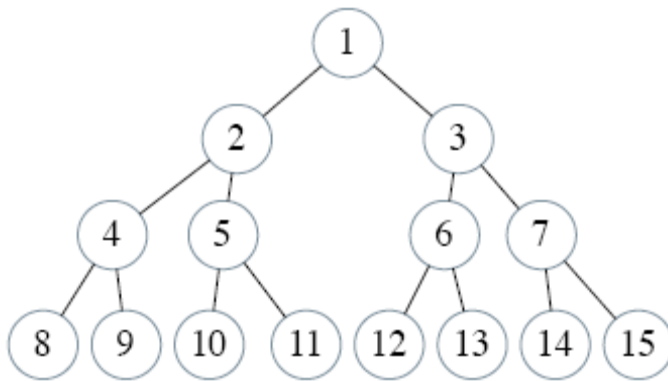
Grading criteria:

Parts (a) is 10 points

Part (b) is 20 points. 10 points for BFS and 10 points for DFS

Part (c) is 20 points. 5 points for search space size and 15 points for the explanation. You should correctly describe the properties of BFS and DFS algorithm.

(a)



(b)

Breadth First Search: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Depth First Search: 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 ...

(c)

The search space is infinite (all positive integers).

I will choose the BFS because it is guaranteed to find the optimal solution (if the optimal solution exists). Although DFS uses less memory, it might get stuck going down a very long path. For example, if the solution is node 6, DFS will explore the whole left sub-tree.