DFS Space

Max. space:
If tree was full (every node had \( b \) children), then we would store \( b \) nodes for every node along a path from top to bottom \( \Rightarrow O(bm) \).
IDS

Try depth-limit = 0, 1, 2, 3, 

ex: A

\[
\begin{array}{c}
A \\
/ \\
B \\
/ \\
C \\
/ \\
D \\
/ \\
E \\
/ \\
F \\
/ \\
G \\
/ \\
H \\
\uparrow \\
\infty
\end{array}
\]

Limit nodes explored

0 A
1 ?A, B, C?
2 ?A-F ?
3 ?A-H ?

seems wasteful (lots of re-exploring but for each limit, only ½ nodes explored at that limit are re-explored)

ex: limit = 2 : A B C D E F re-explored

---

Modify BFS (small ex.)

Frontier cur-state
A (0) A
B(1) C(4) D(4) B
C(4) D(4) E(4) C
Uniform Cost Search (UCS) ex

Frontier | Exp. | cur. state
---------|------|---------
A, B, C | A    | A, B, C |
D, E, F | F    | F, E, H |
G, H, I | G    | G, H, I |

Goal: G

Order explored: a b c d e
Greedy Best First Search on Route-Finding

\[ f(n) = h(n) \] chooses G first.