Reminder
- We have class on Friday, 9/11.
  K-Day starts at noon.

Previous classes
- We defined a unified algorithm for uninformed search strategies: BFS, DFS, DLS, IDS
- We evaluated them along four dimensions:
  * completeness
  * time complexity
  * space complexity
  * optimality
- We used the following parameters:
  b: branching factor (finite)
  d: depth of the goal
  m: maximum depth of the tree
  l: depth limit
- Time complexity is exponential for all, these are hard problems
- IDS brings together the good parts of BFS and DFS by adding the cost of repetitions.
  The asymptotic cost remains the same.
  IDS is complete and optimal and has linear space complexity.

Main takeaway:
Always know the properties of an algorithm very well and be able to clearly identify the changes to the properties when variations are made to the algorithm.

This lecture
- Uniform cost search
- Tree search and graph search algorithms
- Heuristic "informed" search
Uniform Cost Search

action costs are variable
(special case of BFS)

\[ \text{cost of actions are 1} \]

![Diagram of a tree with nodes and edges labeled with costs and actions.]

BFS:
- A, B, C
- Order does not matter, all are equal.

UCS:
- Sorted to facilitate picking up the lowest-cost one first.

Keep bookkeeping information in a node:
- state, depth (cost), path
- "pointer" to the parent

A: SA    F: SCF
"It will work just like BFS."
"If all the costs are 1 it is exactly BFS."

It should be optimal, completeness, time complexity, space complexity, optimality still exponential, similar to BFS.

When to do the goal test?
- When a node is generated
- When a node is expanded

BFS
- If we do the goal test, do we generated find the optimal goal?
  - Yes

What about uniform cost search?

- Optimal goal
- Suboptimal goal
- Optimal goal

A
B 2
1
100
D

C
D 100

E
F 1

G

Optimal goal

Checked generated will be returned to the goal.
Heuristic search

Informed search

Guess (heuristic) of where the solution lies

A*:

T

S

Z

329

253

374

Heuristic

"best" option

greedy

Best-first search

A*