Artificial Intelligence

Lab 5

Informed Search

Agenda

Uninformed vs. informed search

What is Informed Search?

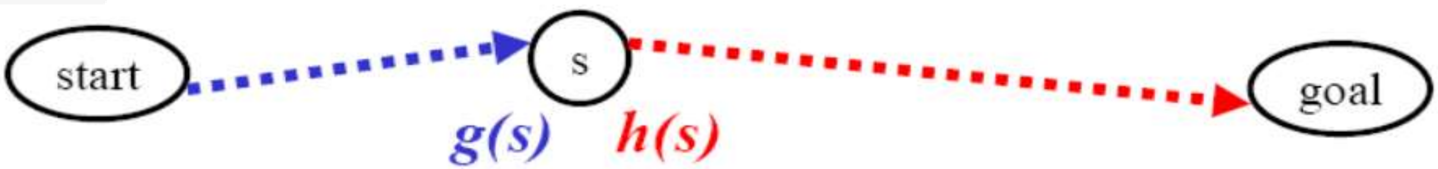
Informed Search

- Heuristic Functions
- A*

Uninformed vs. informed search

- **Uninformed search** (BFS, UCS, DFS)
  - Knows the actual path cost g(s) from start to a node s in the fringe, but that's it.

- **Informed search**



  - Also, has a heuristic value h(s) that represents the cost from start node to goal node. ('h'= heuristic, non-negative)
  - Can be much faster than uninformed search.

Informed Search (Heuristic Search)

- Most informed search algorithms include as a component of f(n) a heuristic function.
  - h(n) is the estimated cost of the cheapest path from the state at node n to a **goal state**.
  - heuristic function are arbitrary, nonnegative, problem specific functions, with one constraint:

    If n is goal node, then h(goal) = 0
- For Example, a heuristic is a **function** tells us **approximately** how far the current state is from the goal state (i.e. smaller numbers are better).

  Heuristic information in search = Hints

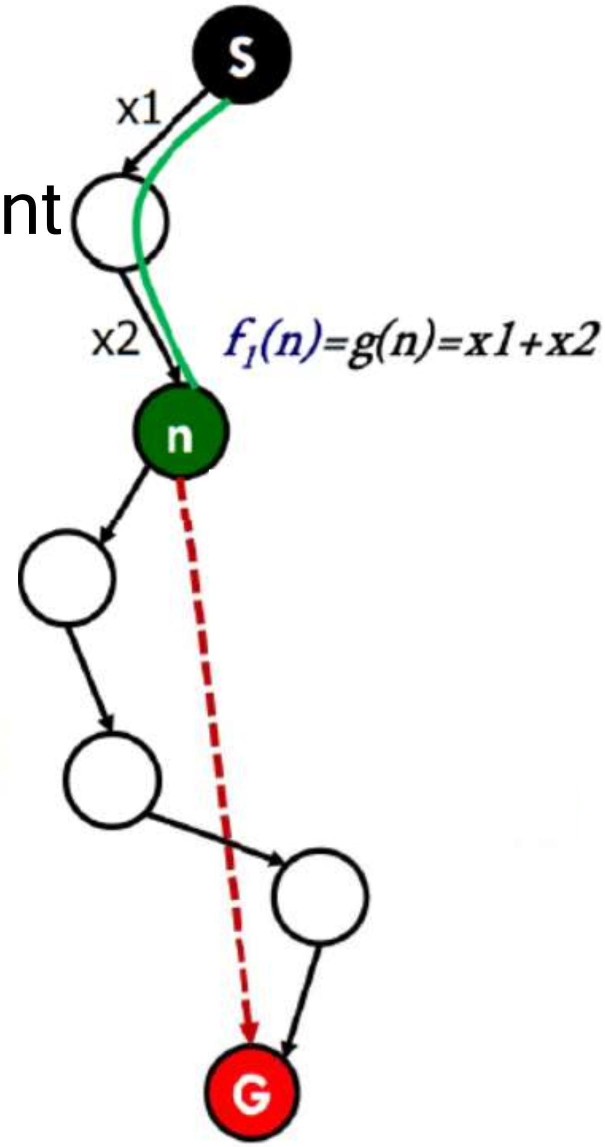
Heuristic Functions

There are 3 approaches to define f(n):

1. f(n) measures the value of the current state (its “goodness” “distance”)

$$f_1(n) = g(n)$$

similar to the uniform cost search (UCS)
where g(n) is the cost to get to n (*from initial state*)



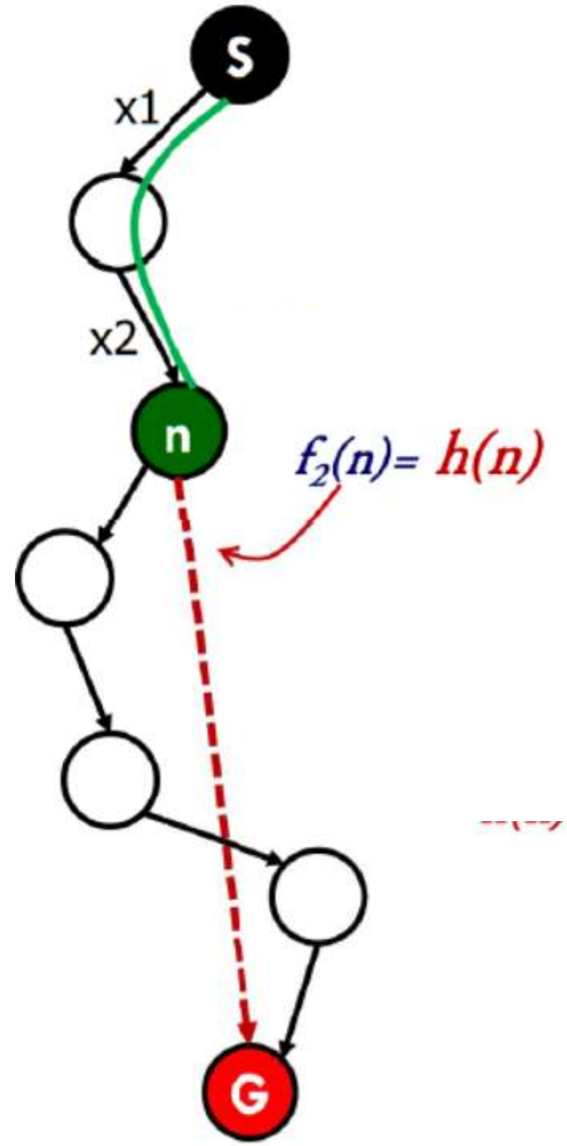
Heuristic Functions

There are 3 approaches to define f(n):

2. f(n) measures the estimated cost of getting to the goal from the current state:

$$f_2(n) = h(n)$$

where $h(n)$ is the estimate of the cost to get from n to a goal

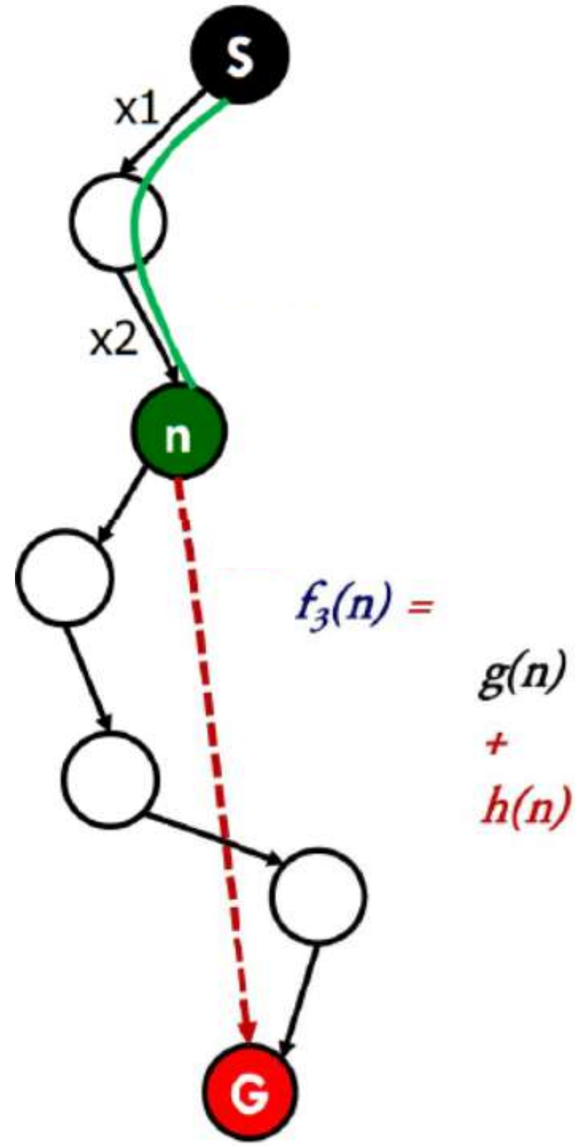


Heuristic Functions

There are 3 approaches to define f(n):

3. f(n) measures the estimated cost of getting to the goal from the current state and the cost to get to n from the initial state.

$$f_3(n) = g(n) + h(n)$$



Informed Search - A*

- Greedy Search minimizes a heuristic h(n) which is an estimated cost from a node n to the goal state. it is **neither optimal nor complete**.
- Uniform Cost Search minimizes the cost g(n) from the initial state to n. UCS is **optimal and complete** but **not efficient**.
- **New Strategy**: Combine Greedy Search and UCS to get an efficient algorithm which is complete and optimal … **A* algorithm**.

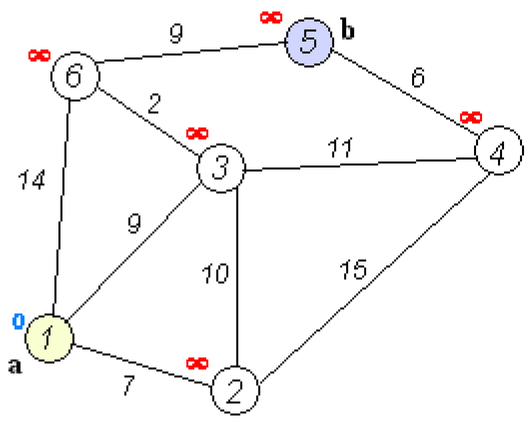
Using Heuristics in A*

- A* uses a heuristic function which combines g(n) and h(n): $f(n) = g(n) + h(n)$.
- **g(n)** is the exact cost to reach node n from the initial state (cost so far up to node n).
- **h(n)** is an estimation of the remaining cost to reach the goal.
- f(n) is the estimated total cost of path through n to goal

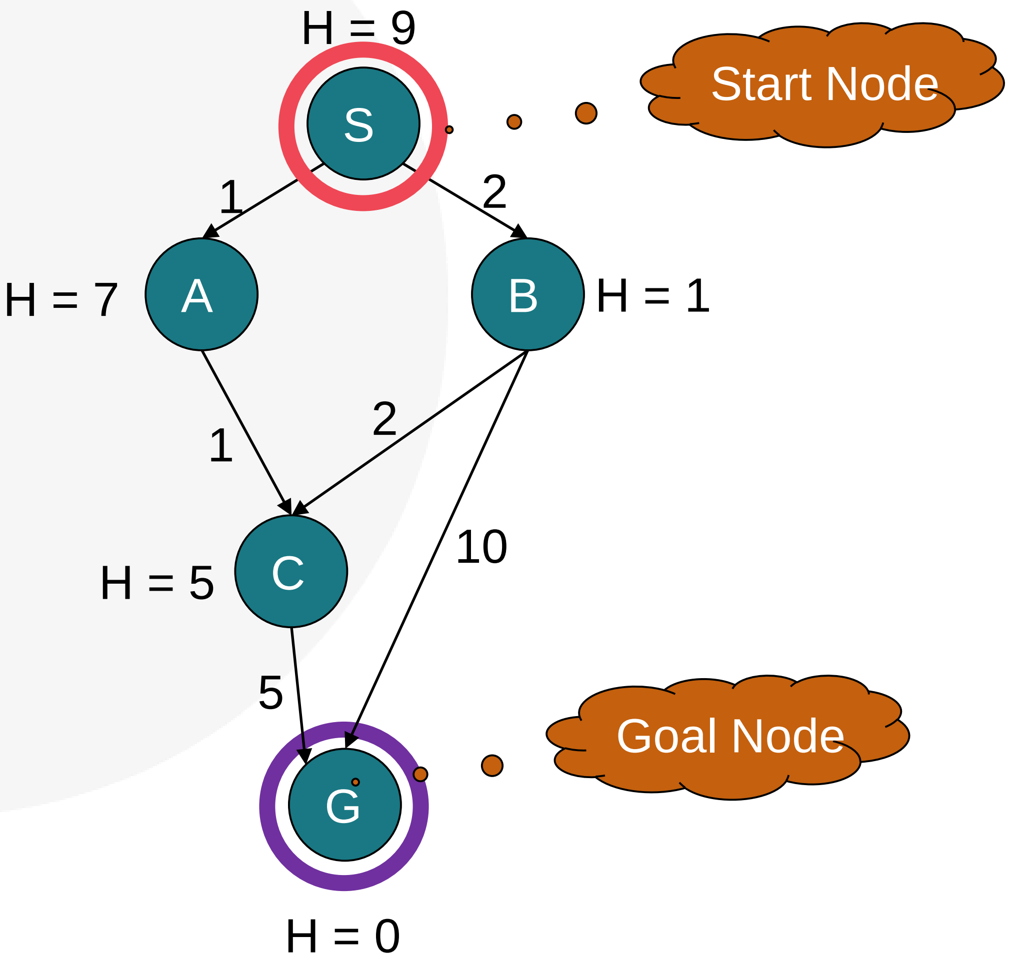
Informed Search – A*

Algorithm A star:

1. Put the start node S on the list called OPEN and empty list for CLOSED
2. If OPEN is empty, exit with failure
3. Remove from OPEN and place on CLOSED a node n for which f(n) is minimum
4. If n is a goal node, exit (trace back pointers from n to S)
5. Expand n, generating all its successors and attach to them pointers back to n. For each successor n' of n
   1. If n' is not on OPEN, estimate h(n'),g(n')=totalOldCost(S->n'), f(n')=g(n')+h(n'), and place it on OPEN and sort OPEN Ascending.
   2. If n' is already on OPEN, then check if h(n') is lower for the new version of n'. If so, then update h(n’) with lower value.
   3. If g(n') is not lower for the new version, do nothing.
6. Goto 2.



Informed Search – A*



Opened	Closed
{9S}	{}
{3B,8A}	{9S}
{8A, 9C,12G}	{9S, 3B}
{7C , 12G}	{9S, 3B,8A}
{7G}	{9S, 3B,8A, 7C}
	{9S, 3B,8A, 7C, 7G}

Traversed Path:

{S, A, C, G}
With cost = 7

Example – Puzzle 8 Game

- Given an initial state of the board, the combinatorial search problem is to find a sequence of moves that transitions this state to the goal state
- The blank space is going to be represented with the number 0.
- The blank space may be swapped with a component in one of the four directions {'Up', 'Down', 'Left', 'Right'}, one move at a time

Initial state =

2	8	3
1	6	4
7		5

Goal State =

1	2	3
8		4
7	6	5

Heuristic Measures the Cost to the Goal

- Assuming that moving one tile in any direction will have 1 unit cost.
- A state X would be better than a state Y if the estimated cost of getting from X to the goal is lower than that of Y
  - because X would be closer to the goal than Y

- 8–Puzzle heuristic Function:

  There are two ways of calculations with ignoring blank node:

  h1: The number of misplaced tiles.

  h2: The sum of the distances of the tiles from their goal positions. By calculating how far the tile from its goal position is to sum the number of horizontal and vertical positions.

Calculating Heuristic

5	4	
6	1	8
7	3	2

Current State

1	2	3
8		4
7	6	5

Goal State

Using H1:

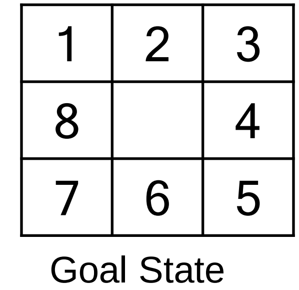
H(Initial State) = 7

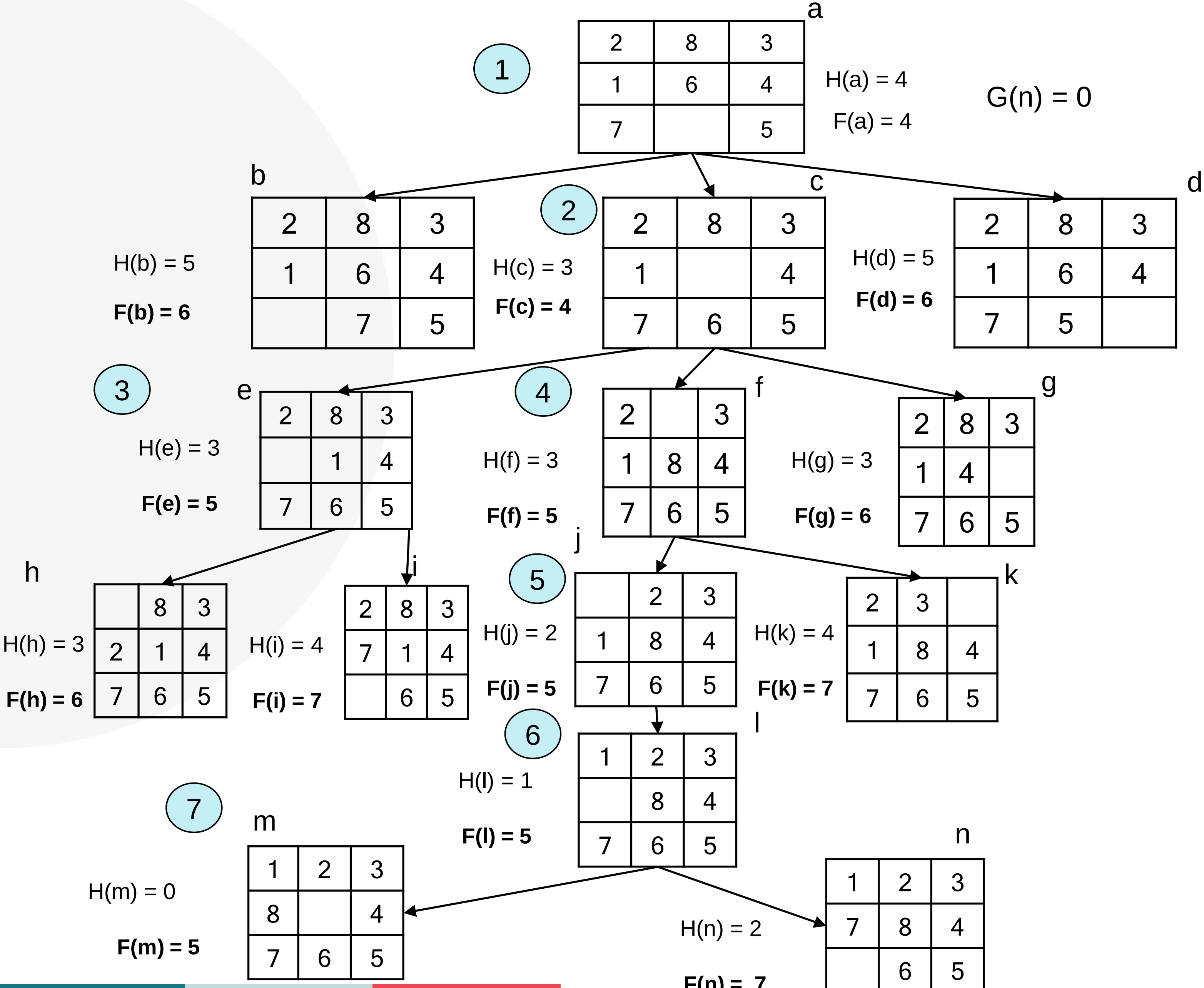
Using H2:

H(Initial State) = 2 + 3 + 3 + 2 + 4 + 2 + 0 + 2 = 18

2 represents how many steps does 1 need to be in his goal position.

8 Puzzle Game – Using A* + H1





Milestone

Milestone 2 will include search algorithms (BFS, DFS, UCS and A*).

It will be announced on course-sites.

Questions?