

6.034 Final Examination

December 17, 2001

Name	
E-mail	

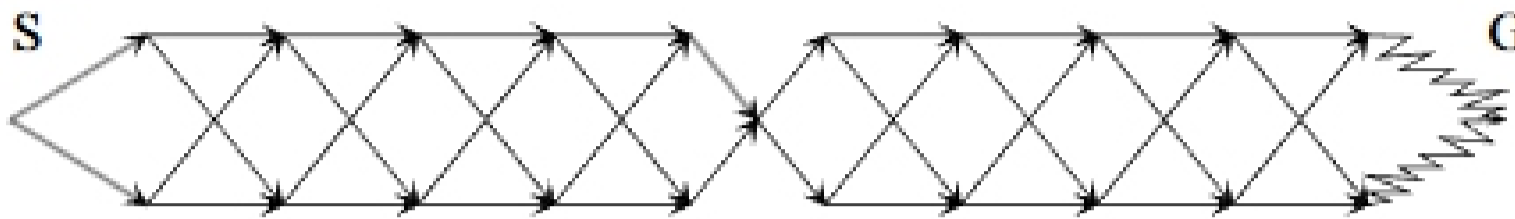
Indicate TA:

Jake Beal	<input type="checkbox"/>
Andy Crane	<input type="checkbox"/>
Jimmy Lin	<input type="checkbox"/>
Andrew Nevins	<input type="checkbox"/>
Doug Ricket	<input type="checkbox"/>
Justin Schmidt	<input type="checkbox"/>

Problem Number	Max	Score	Grader
Problem 1	16		phw rcb psz jb ac jl an dr js
Problem 2	16		phw rcb psz jb ac jl an dr js
Problem 3	20		phw rcb psz jb ac jl an dr js
Problem 4	20		phw rcb psz jb ac jl an dr js
Problem 5	18		phw rcb psz jb ac jl an dr js
Problem 6	10		phw rcb psz jb ac jl an dr js
Total	100		

Problem 1: Search in a weird city (16 points)

You have decided to look for a job driving a cab. The recruiter, an MIT graduate, asks you to solve some search problems for streets laid out as shown by the lines in the figure.



The map is drawn to scale, but note that the final, jagged streets converging on G can be assumed to be 100 times longer than any other street. All streets are one-way, left to right. In the absence of any other criterion, or a tie, you are to follow the option closer to the top of the page.

Your job is to find a path from the start node, S, to the goal node, G, according to various conditions. Note that the total number of nodes is 23.

You may express your answers either as numbers (e.g. 8) or variable-free expressions (e.g. $2^0 + 2^1 + 2^2 + 2^3$) or both. You get full credit if either is right. Also, you will increase your partial credit if you correctly note when two answers should be the same, even if both are wrong.

Part A: Depth-First Search, with backtracking (2 points)

You are not to use a visited (enqueued) list or expanded list. Search is to terminate when a path that reaches the goal appears at the front of the search queue. Indicate the number of nodes expanded.

12

Part B: Breadth-First Search (2 points)

You are to use a visited (enqueued) list. Search is to terminate when a path that reaches the goal appears at the front of the search queue. Indicate the number of nodes expanded.

22 (all nodes other than the goal)

Part C: Branch and Bound (2 points)

You are to find the optimal path, using neither an expanded list, nor an admissible heuristic. Indicate the number of nodes expanded.

$1024 + 32 + 1024 - 1 = 2079$

Part D: A* (4 Points)

You are to find the optimal path, using an expanded list, and the distance to the goal as an admissible heuristic. Indicate the number of nodes expanded.

22 (expanded list saves the day)

Part E: Bidirectional Breadth-First Search (4 Points)

You are not to use a visited (enqueued) list or expanded list. Search from the left is interdigitated with search from the right. That is, one search starts from the left and produces all paths of depth 1; then, another search starts from the right, goes against the arrows, and produces all paths of depth 1. The two searches then alternate, adding additional layers to their search trees, until an expansion produces a path from the left that shares a node with a path from the right. Indicate the number of nodes expanded.

$63 + 63 + 32 + 1$

Part F: Comparison (2 Points)

Evidently, bidirectional breadth-first search does less work than unidirectional breadth-first search would have done in Part E. In fact, the ratio of unidirectional nodes expanded to bidirectional nodes expanded is on the order of (circle best answer).

- 2

Correct answer follows. Bidirectional search cuts exponent in half, dramatically reducing search effort.

- $b^{d/2}$
- b^d
- $2 \times b \times d$

Where b is the branching factor and d is the total depth as seen by unidirectional search.