

Exam 1 Practice

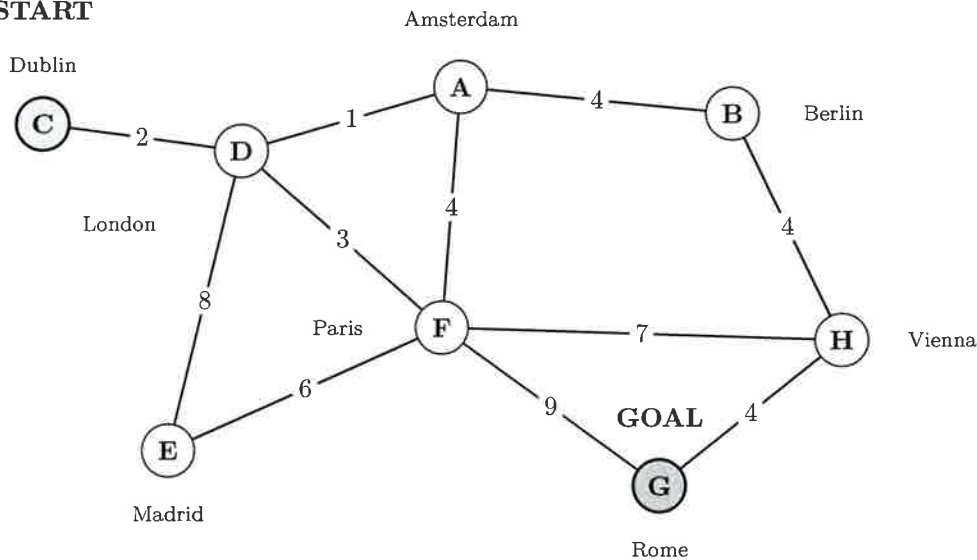
Name: _____

Date: _____

1. Search order

Start at **C (Dublin)** and search for **G (Rome)** in the undirected graph below. Successors enter the frontier alphabetically. Visits and the goal test occur on removal from the frontier; repeated visits are skipped. Do not add already visited states. Iterative deepening uses limits 0, 1, ... and path-based cycle checking.

START



Write the order of visited nodes for each algorithm below.

Depth-first search:

C, D, F, H, G

Breadth-first search:

C, D, A, E, F, B, G

Uniform cost search:

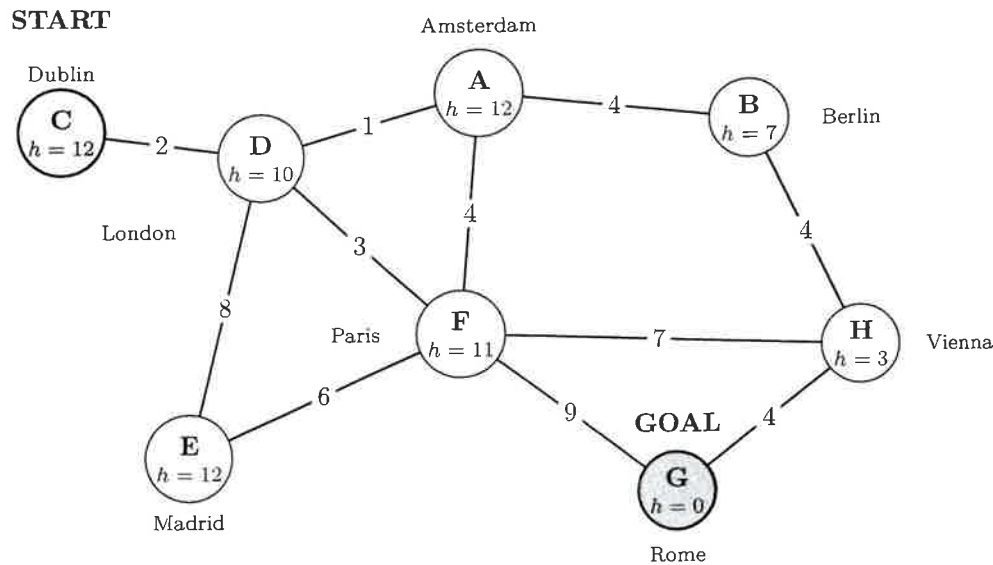
C, D, A, F, B, E, H, G

How many times would node D be visited in iterative deepening before the goal is reached?

3: The goal is a depth of 3, so I D would visit D 3x

2. Heuristic

The same graph is shown below with a heuristic $h(n)$ estimating the remaining cost to goal G. Edge weights are unchanged.



(a) Is this heuristic admissible? Explain why or why not.

No, $h(F)$ over estimates cost
(11 v.s. 9)

Admissible: Never
over estimates goal cost

(b) Is this heuristic consistent? Explain why or why not.

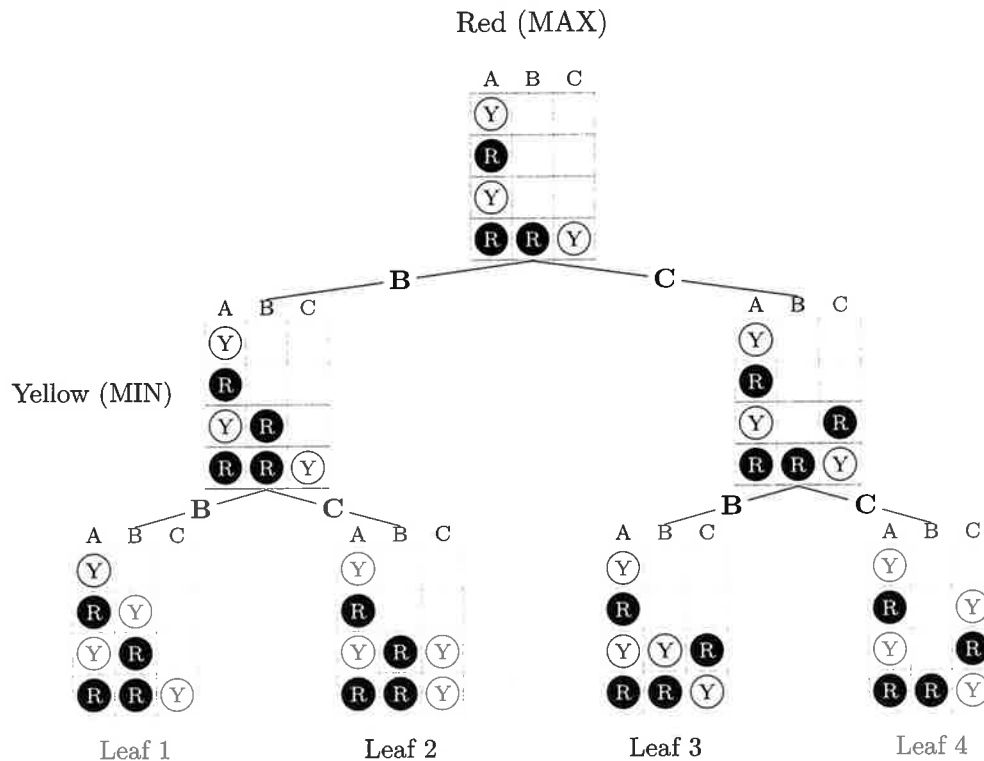
No, $h(A) > h(B) + \text{Cost}(A \rightarrow B)$
12 7 4

Consistent:
 $h(s) \leq h(s^*) + \text{Cost}(s \rightarrow s^*)$
i.e. cannot drop by more than
distance

3. Evaluation functions and bounded minimax

In Connect Three, players alternate dropping a piece into a column; it falls to the lowest empty cell. A player who gets three of their color in a row horizontally, vertically, or diagonally wins.

Red is MAX (+1); Yellow is MIN (-1).



(a) Propose an evaluation function for nonterminal positions. Describe it in words, with higher values indicating better positions for Red.

Number of 2-in-a-row for Red, minus 2-in-a-rows for yellow

(b) Write the value your function gives each of the four leaves.

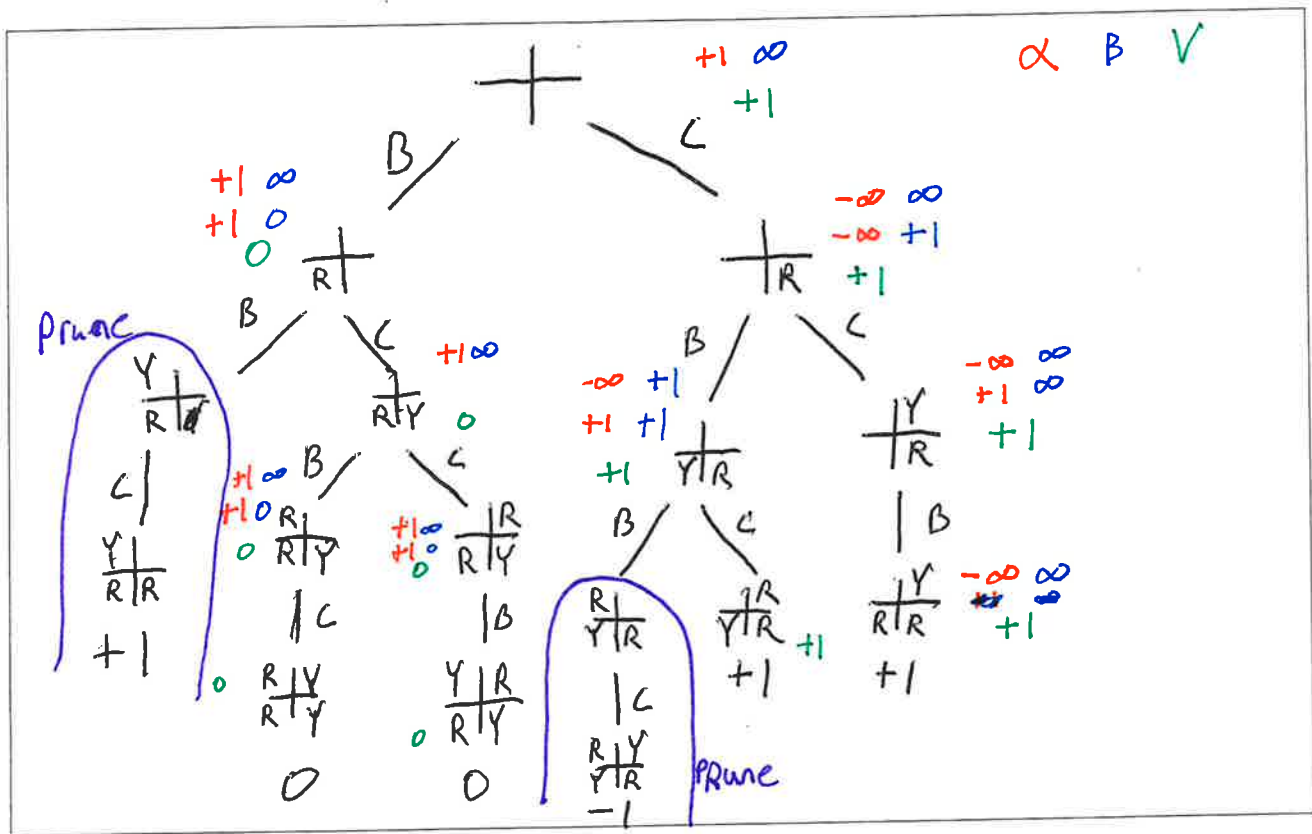
$$\begin{array}{cccc} \text{Leaf 1} & \text{Leaf 2} & \text{Leaf 3} & \text{Leaf 4} \\ 3-2=1 & 3-1=2 & 2-2=0 & 2-0=2 \end{array}$$

Note: This is not a great eval function...

4. Minimax and alpha-beta pruning

At leaf 3, Red moves next. Values are +1 for a Red win, -1 for a Yellow win, and 0 for a draw.

(a) Draw the complete minimax tree rooted at leaf 3, including its values. What is the value of leaf 3?



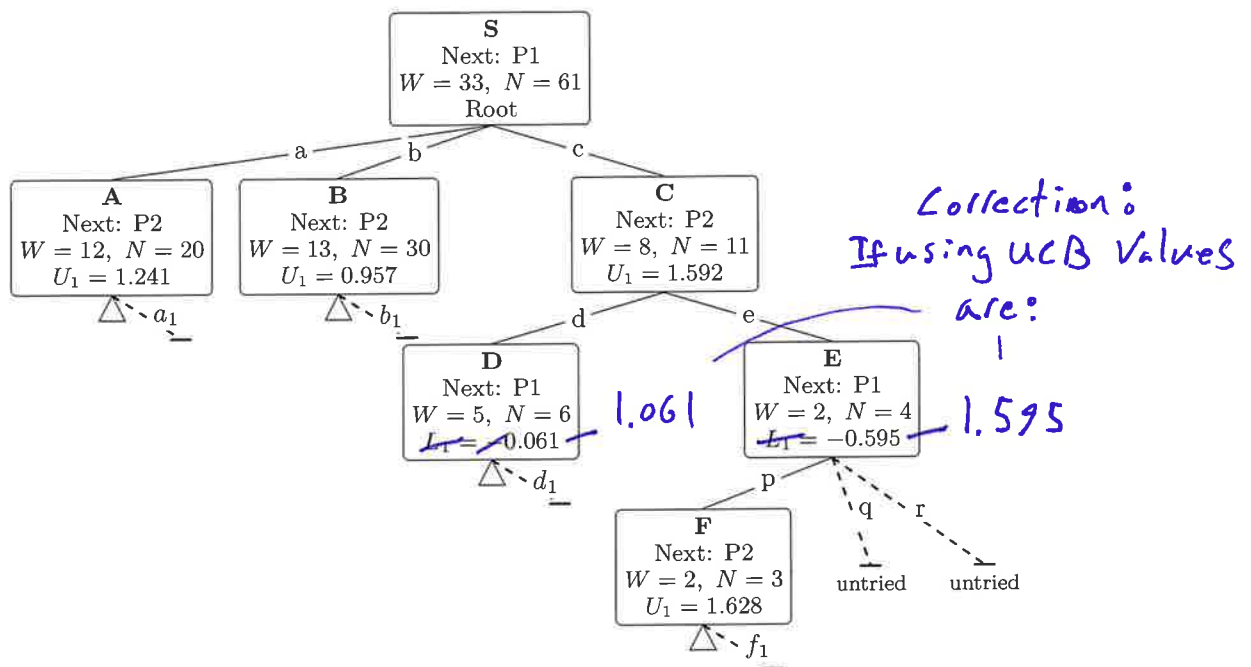
Value of leaf 3: +1

(b) Which nodes would alpha-beta pruning skip? Assume that you always try move C before B

Shown above

5. Monte Carlo tree search

Consider one MCTS iteration from S . W always counts **Player 1 wins**; N counts visits. There are no draws. Break ties using alphabetical order.



Triangles denote omitted descendants; dashed cutoffs denote untried actions.

(a) What path does selection follow, and at which node does it stop?

$S \rightarrow C \rightarrow E$, stops at E

(b) Where is the new node X added, and which action creates it?

At q, Expansion

Rollout result: Player 1 wins.

5. MCTS continued

(c) Update the table after backpropagating a Player 1 win. Give selection bounds to three decimal places. The exploration bonuses below may be used.

Parent visits $N(u)$	Child visits $N(v)$	Exploration bonus $\sqrt{2 \ln N(u)/N(v)}$
62	20	0.642
62	30	0.525
62	12	0.829
12	6	0.910
12	5	0.997
5	3	1.036
5	1	1.794

Node	Updated wins W	Updated visits N	Updated bound (U_1 or L_1)
S	34	62	—
A	12	20	1.842 $V = 1 + \frac{12-8}{20}, N=62, n=20$
B	13	30	1.391 $V = 1 + \frac{13-17}{30}, N=62, n=30$
C	9	12	2.329 $V = 1 + \frac{9-3}{12}, N=62, n=12$
D	5	6	1.243 $V = 1 + \frac{5-1}{6}, N=12, n=6$
E	3	5	1.797 $V = 1 + \frac{3-2}{5}, N=12, n=5$
F	2	3	2.369 $V = 1 + \frac{2-1}{3}, N=5, n=3$
X	1	1	3.794 $V = 1 + \frac{1-0}{1}, N=5, n=1$

(d) Which nodes have unchanged wins and visits but a changed selection bound? Why?

A, B, D, F, Parent visits increase, changing the bound

6. Local search: job scheduling

A single machine processes the following jobs. Times are in hours. Each job incurs a waiting cost from its arrival until processing begins. Jobs cannot be interrupted. Given a job order, each job starts as soon as it has arrived and the preceding job has finished. The machine is available at time 0.

Find an order that minimizes total waiting cost.

Job	Arrival time	Duration	Waiting cost per hour
A	0	3	\$2
B	1	1	\$6
C	2	2	\$3
D	3	1	\$5
E	0	2	\$1

(a) Formulate this as a local search problem: describe a state representation, successor function, and objective function.

State: An order to schedule jobs

Successor func: Swap 2 jobs in the order

Objective: Total waiting cost (assume jobs scheduled before arrival cannot start until arrival)

(b) What is the total waiting cost for the order A, B, C, D, E?

A: 0 wait $\rightarrow$ \$0

D: starts at 6 Cost = $5(6-3) = 15$

B: starts at 3 Cost = $6(3-1) = 12$

E: starts at 7 Cost = $1(7-0) = 7$

C: starts at 4 Cost = $3(4-2) = 6$

Total = $12 + 6 + 15 + 7 = 40$

(c) Give a neighboring order with a lower cost, using your successor function. What is its cost?

Swap D and C $\rightarrow$ A, B, D, C, E

D: starts at 4 Cost = $5(4-3) = 5$

E: starts at 5 Cost = $3(5-2) = 9$

total = $12 + 9 + 5 + 7 = 33$