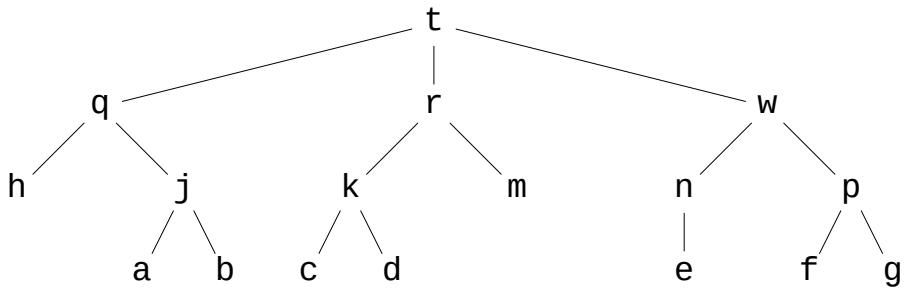
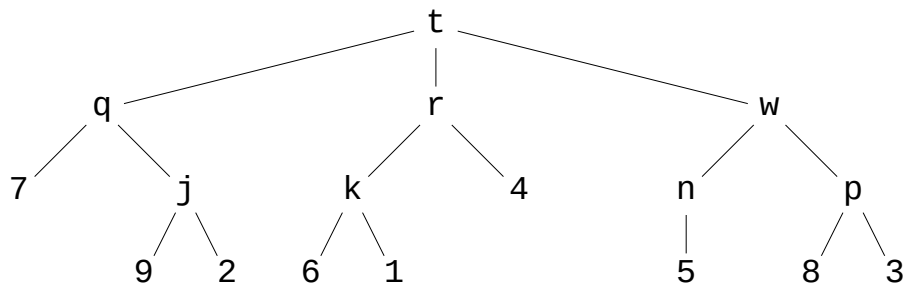
a 2-player game tree



with leaf scores (say for first player P1)

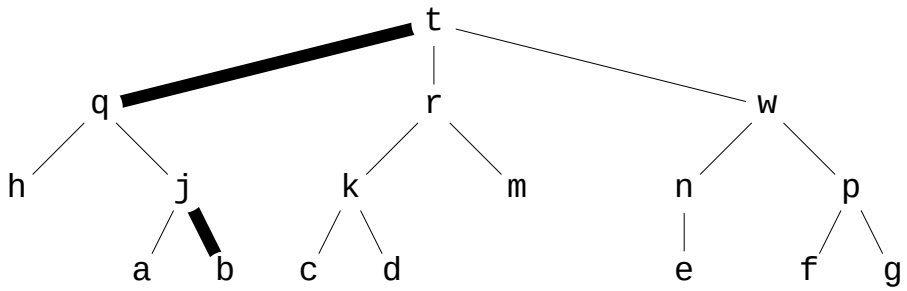


each game is a path from root to a leaf

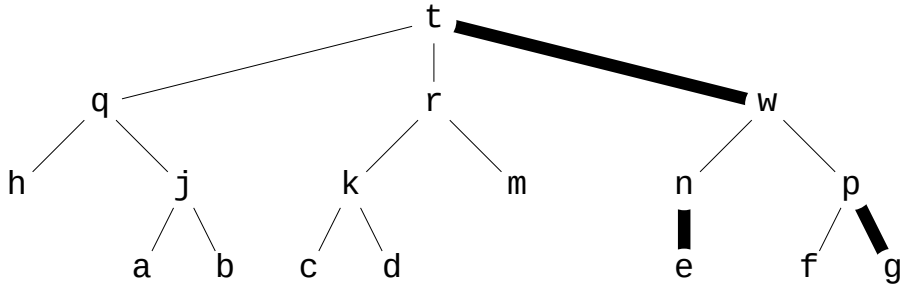
e.g. game t-q-h P1 score 7

e.g. game t-r-k-d P1 score 1

a 1st-player strategy 1A



a 1st-player strategy 1B

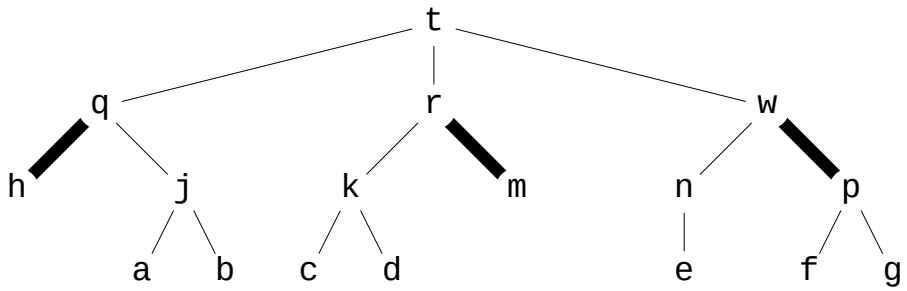


a strategy shows each possible choice a player makes

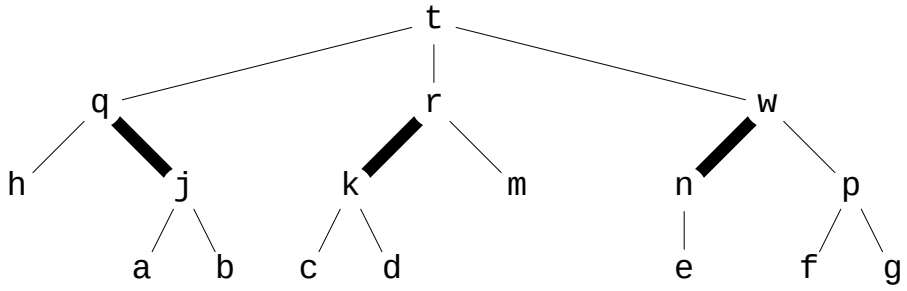
1A: at root, choose q; at h, no choice needed; at j, choose b

1B: at root, choose w; at n, choose e; at p, choose g

a 2nd-player strategy 2C



a 2nd-player strategy 2D



how do these strategies do against each other?

	2C	2D
1A	t-q-h 7	t-q-j-b 2
1B	t-w-p-g 3	t-w-n-e 5

P1 picks A: min score for P1 is 2

P1 picks B: min score for P1 is 3

P1 wants to maximize her min score against {C, D}? pick B

how do these strategies do against each other?

	2C	2D
1A	t-q-h 7	t-q-j-b 2
1B	t-w-p-g 3	t-w-n-e 5

P2 picks C: max score for P1 is 7

P2 picks D: max score for P1 is 5

P2 wants to minimize P1's max score against {A, B}? pick D

minimax

P1: for each P1-strat S, find S's min score, against all possible opponent strategies

then, over all choices for S, pick maximizing

P2: for each P2-strat T, find T's min opponent-score, over all possible opponent strategies

then, over all choices for T, pick maximizing

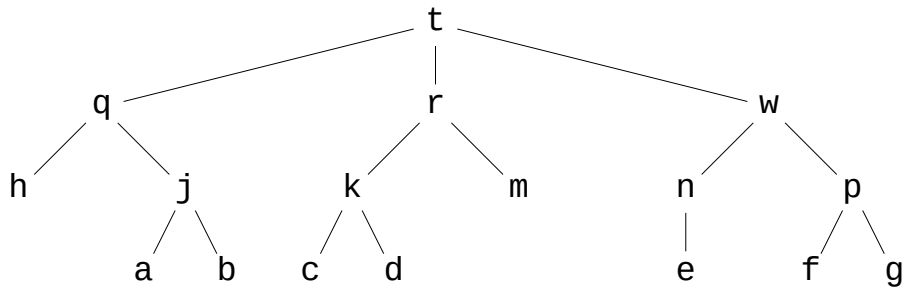
min-then-max $\Rightarrow$ minimax

for this game, P1 has 6 possible strategies, P2 has 8

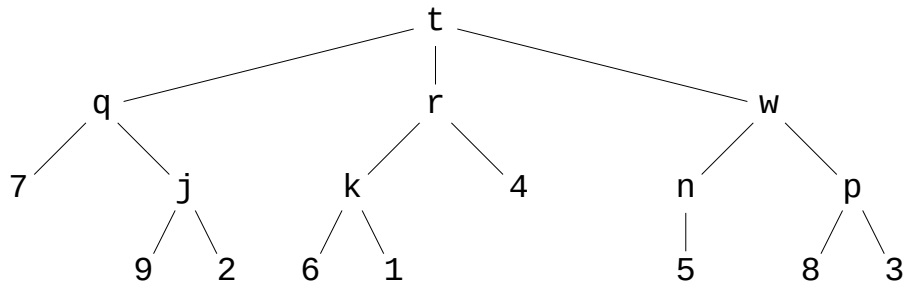
good news: we don't have to consider them individually

use minimax algorithm

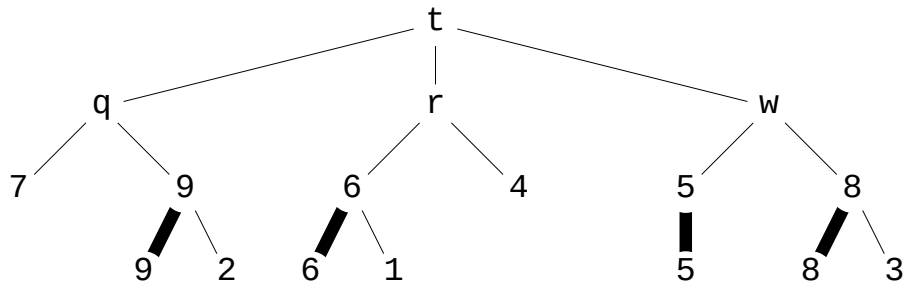
compute minimax values bottom-up



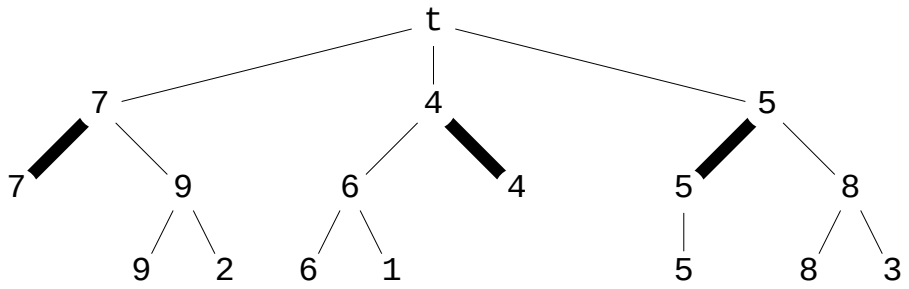
compute minimax values bottom-up



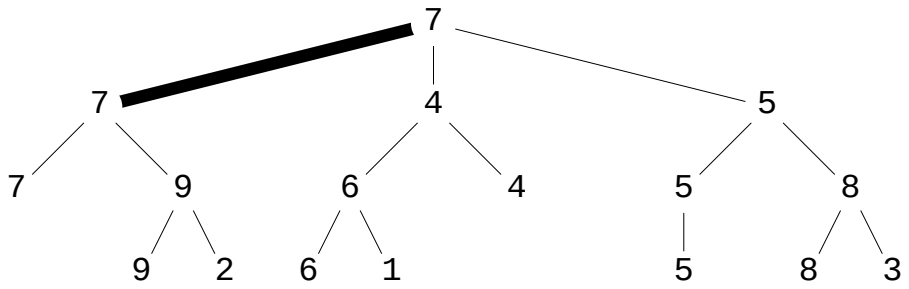
compute minimax values bottom-up



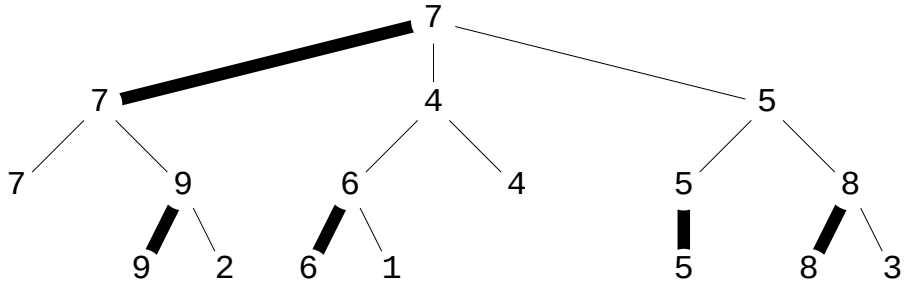
compute minimax values bottom-up



compute minimax values bottom-up



P1 minimax strat



P2 minimax strat