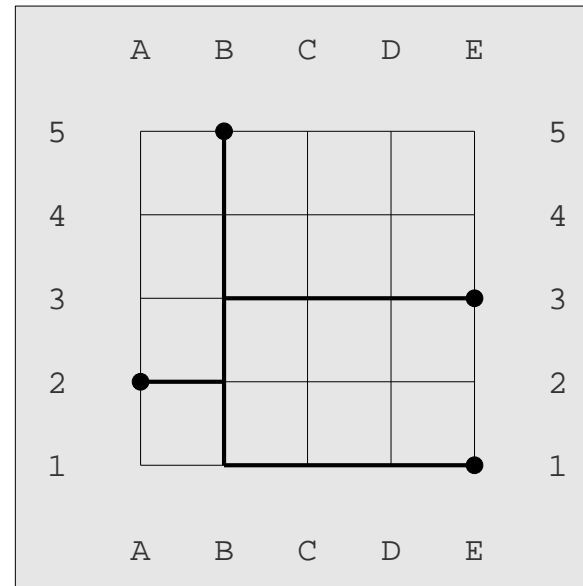
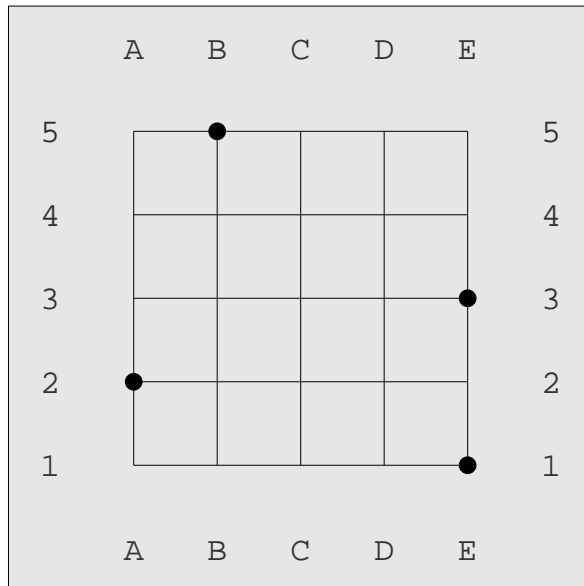


steiner tree puzzles (courtesy Bonn Discrete Math Inst.)

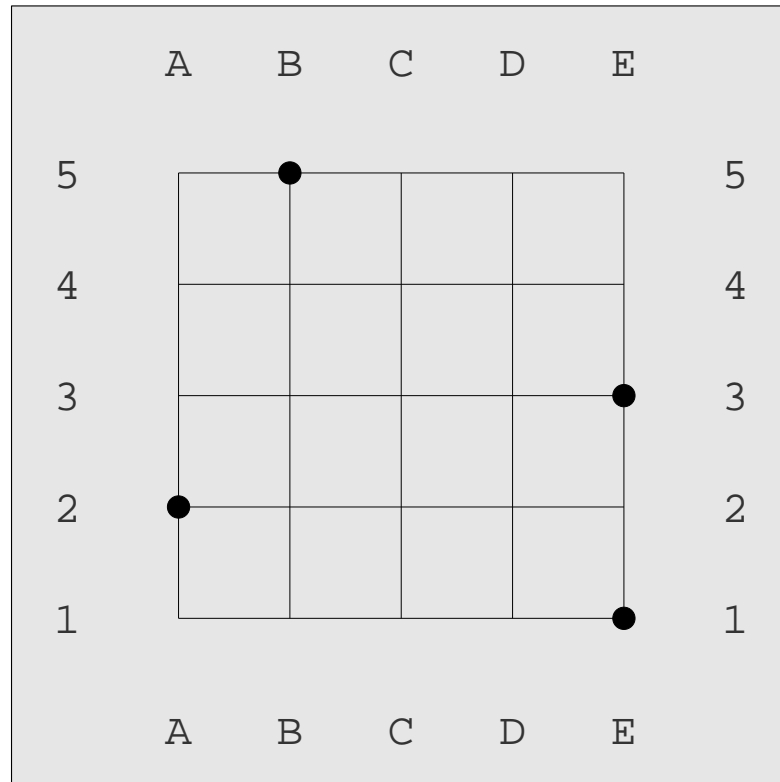
*steiner tree* joins all given nodes (a.k.a. pins)

steiner tree *cost* is sum of lengths of tree edges

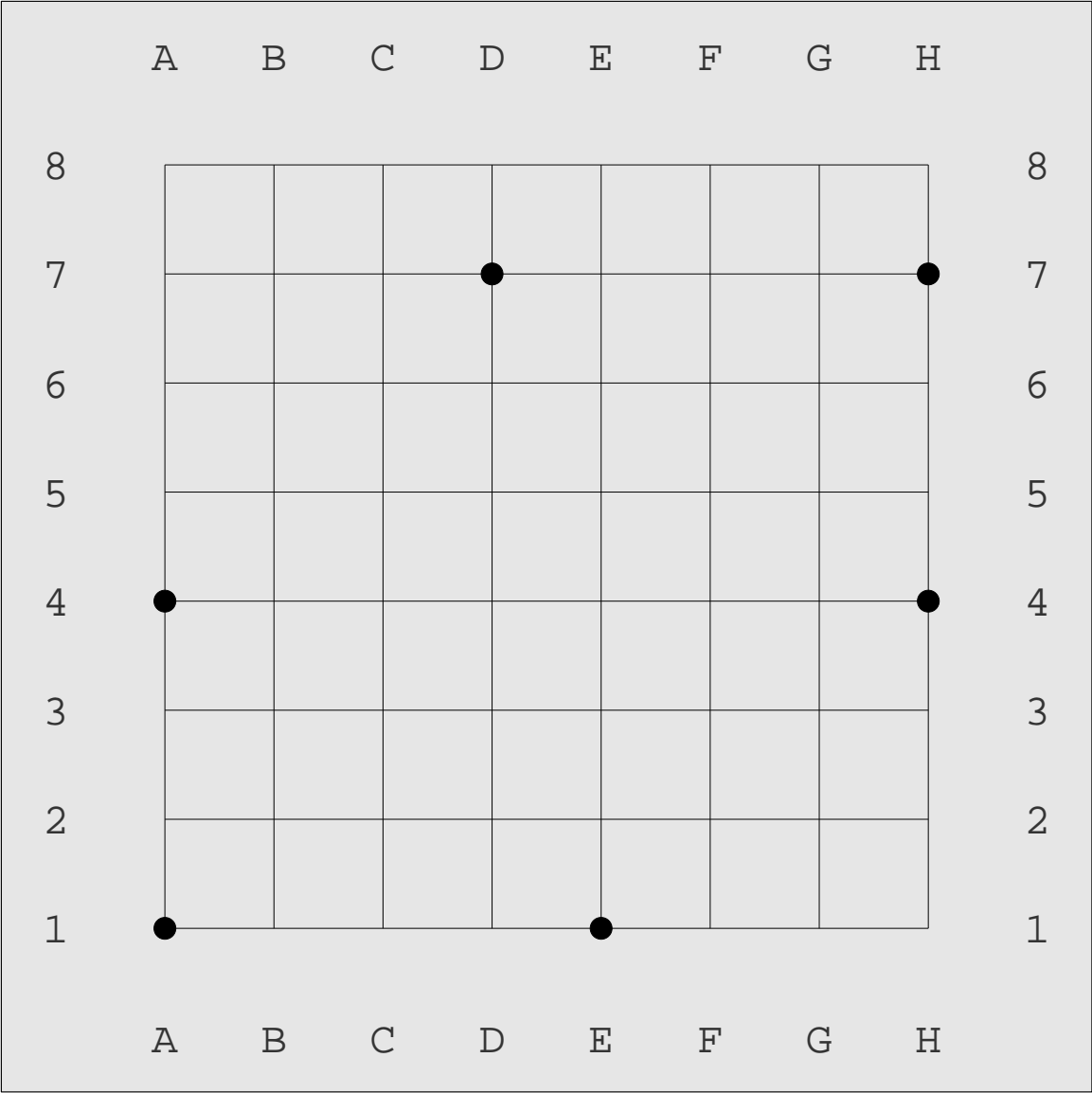
example: this tree has cost 11, can you do better?



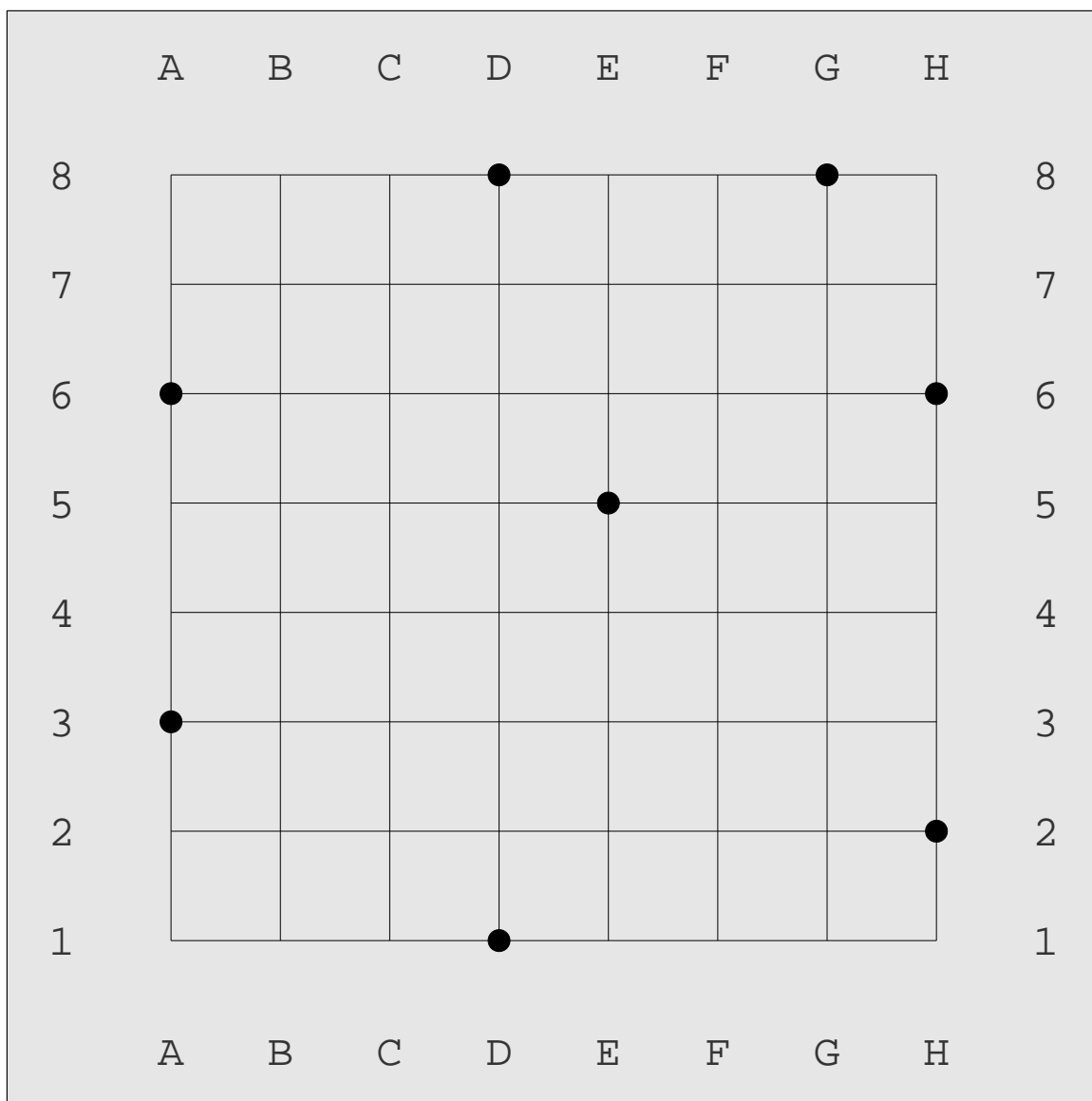
for each puzzle, find a min-cost steiner tree



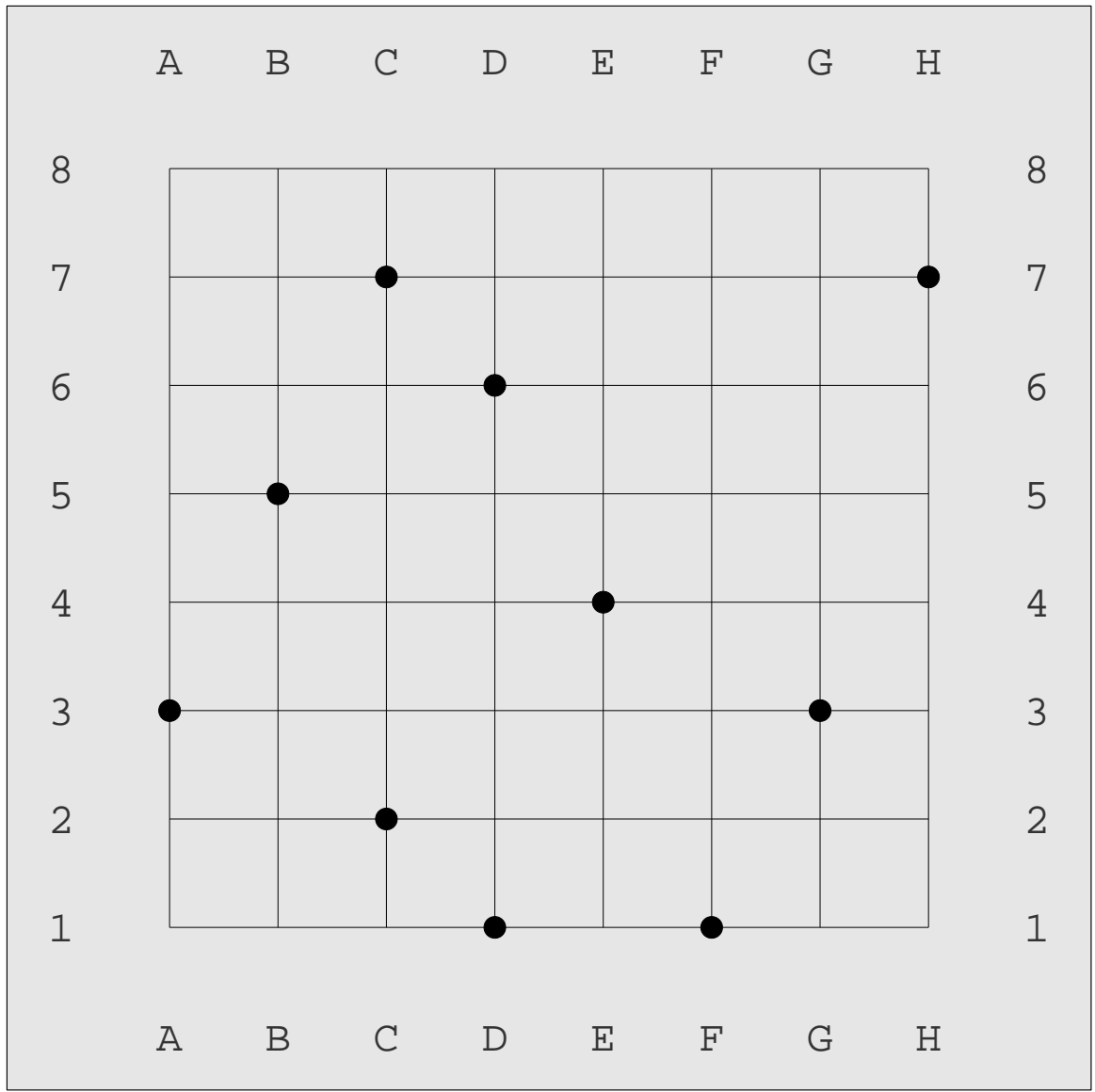
easy 4 pins, cost 9



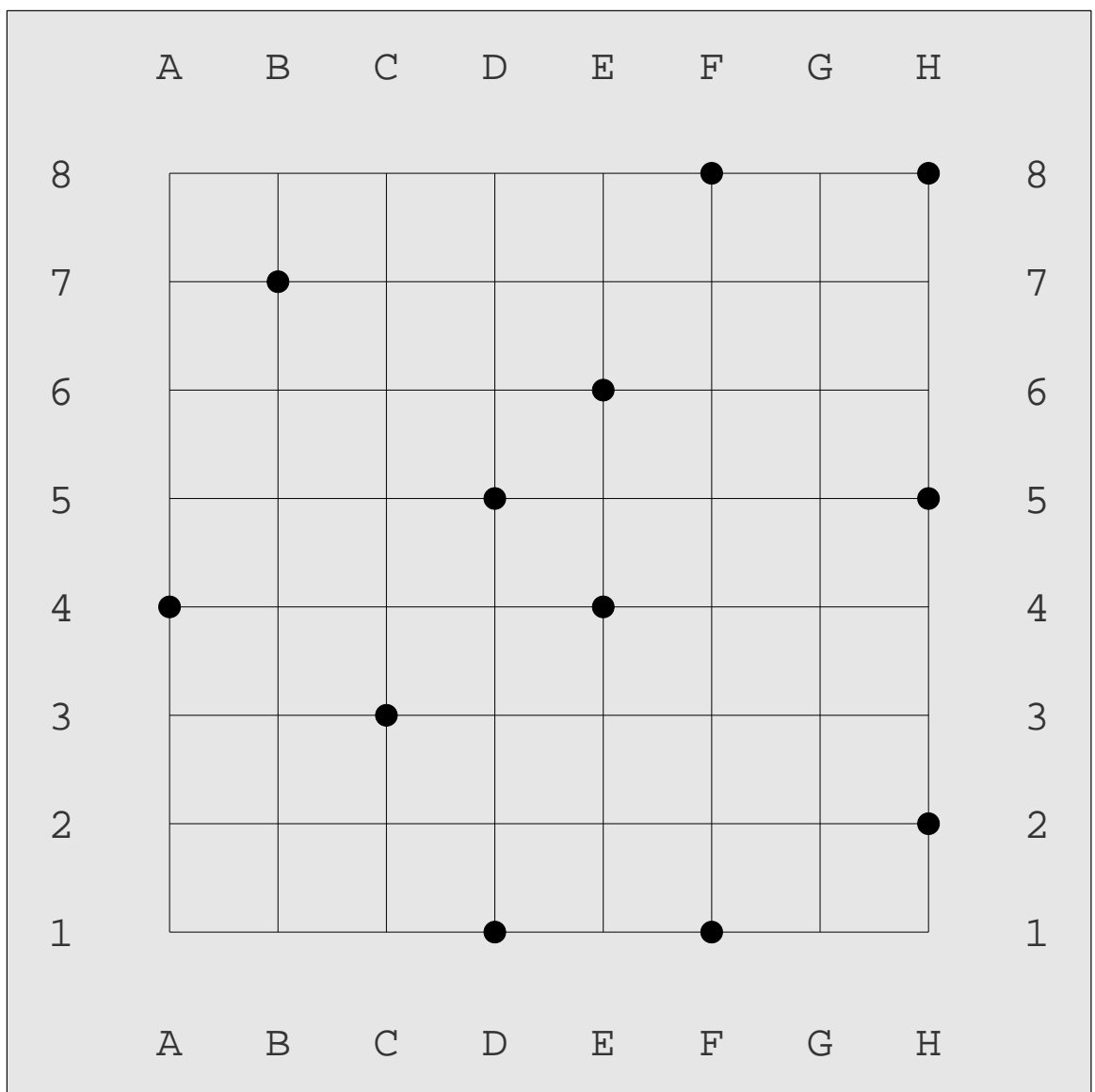
easy 6 pins, cost 19



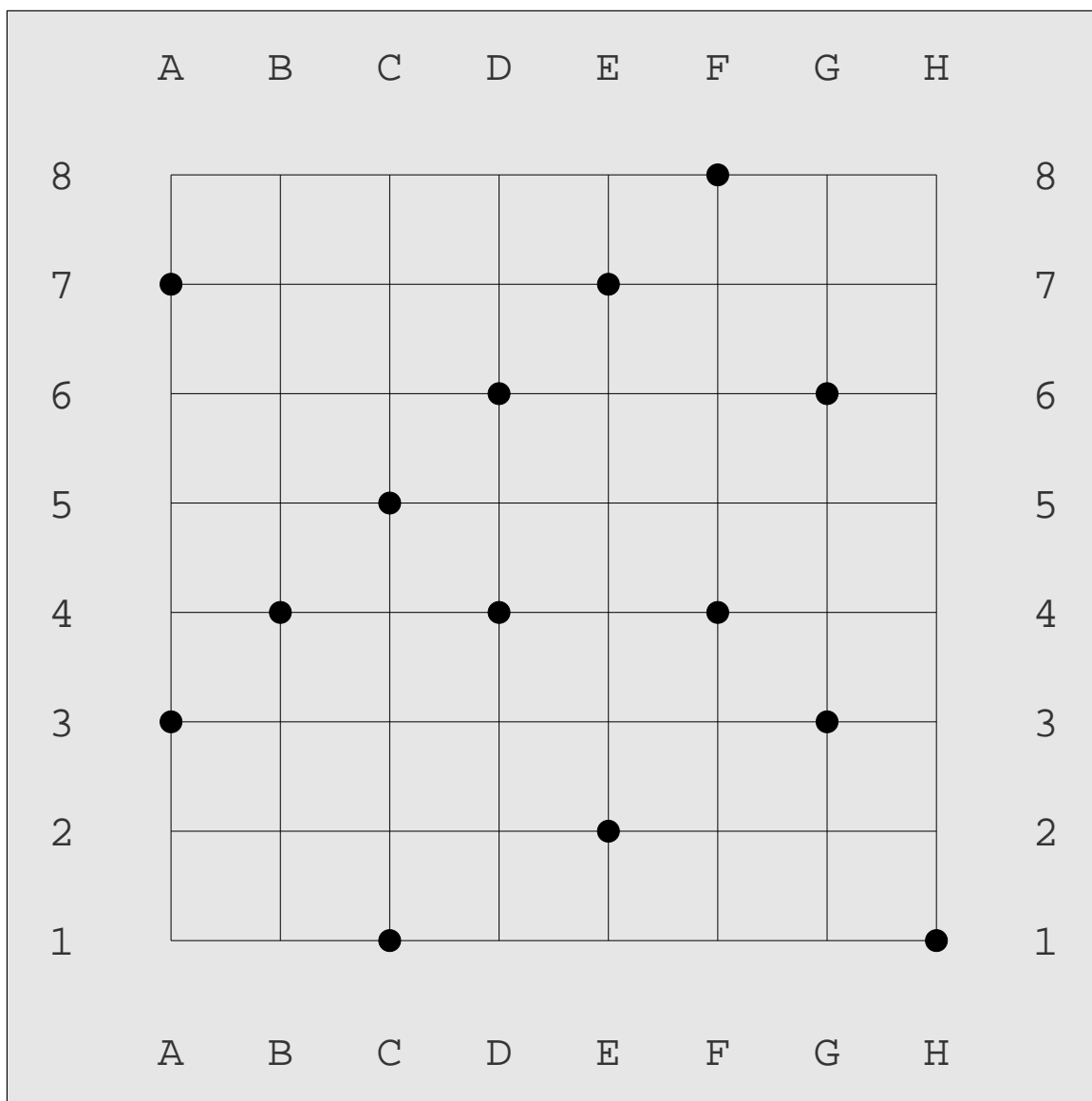
easy 8 pins, cost 23



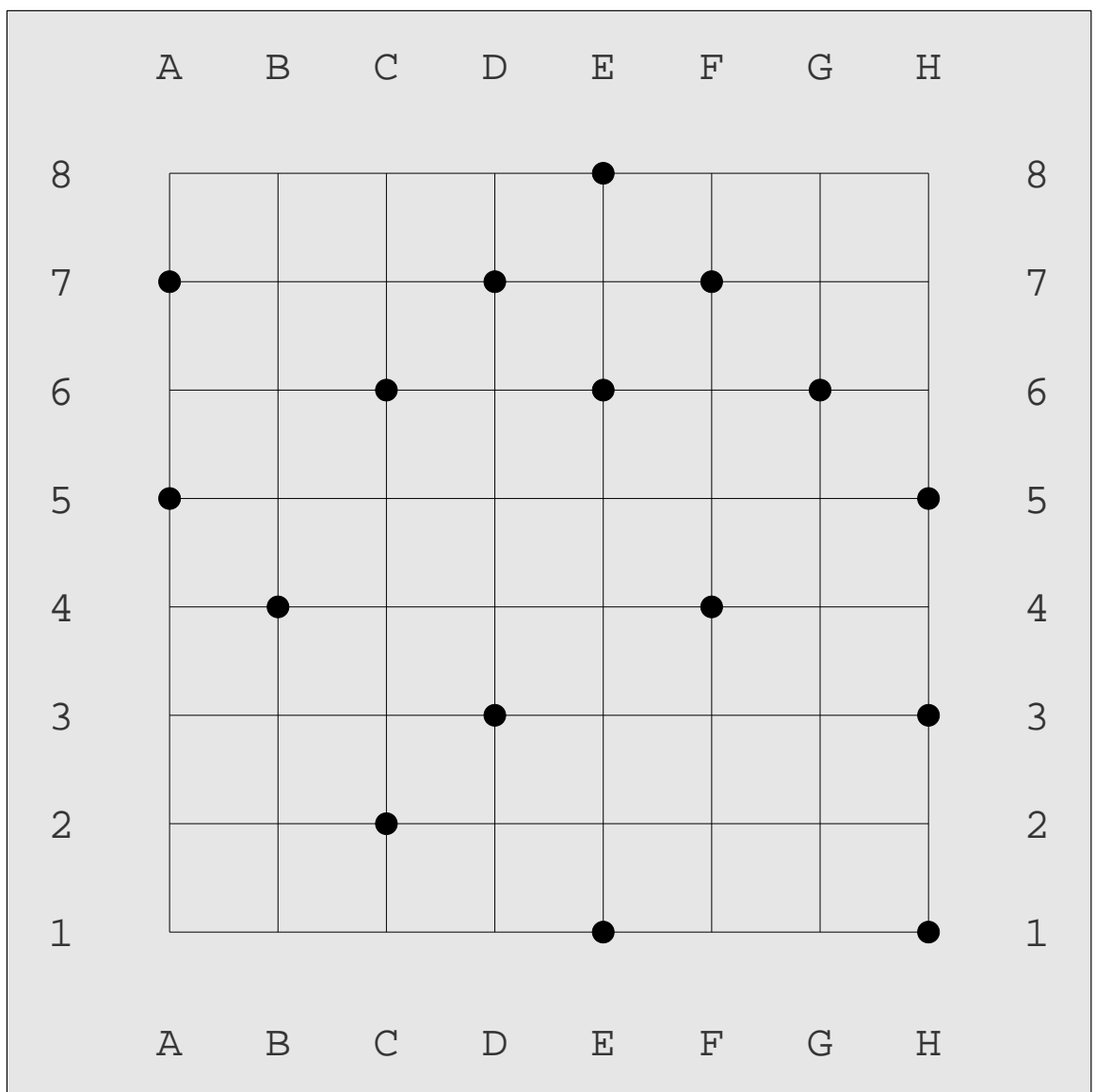
**medium 10 pins, cost 22**



hard 12 pins, cost 26



harder 14 pins, cost 27



harder 16 pins, cost 28



## steiner tree approx

- from input  $G$ , create pins-graph  $P(G)$
- $P(G)$  node set: pins of  $G$
- $P(G)$  edge set: all pairs of pins of  $G$
- for each edge of  $P(G)$ , cost is min-cost path in  $G$
- find mst  $M$  of  $P(G)$
- claim:  $G$  has steiner tree with cost  $\leq \text{cost}(M)$
- proof:
- for each edge  $(x, y)$  of  $M$ , add corresponding min-cost path in  $G$   
(on the path, don't add any edge that would create a cycle)

