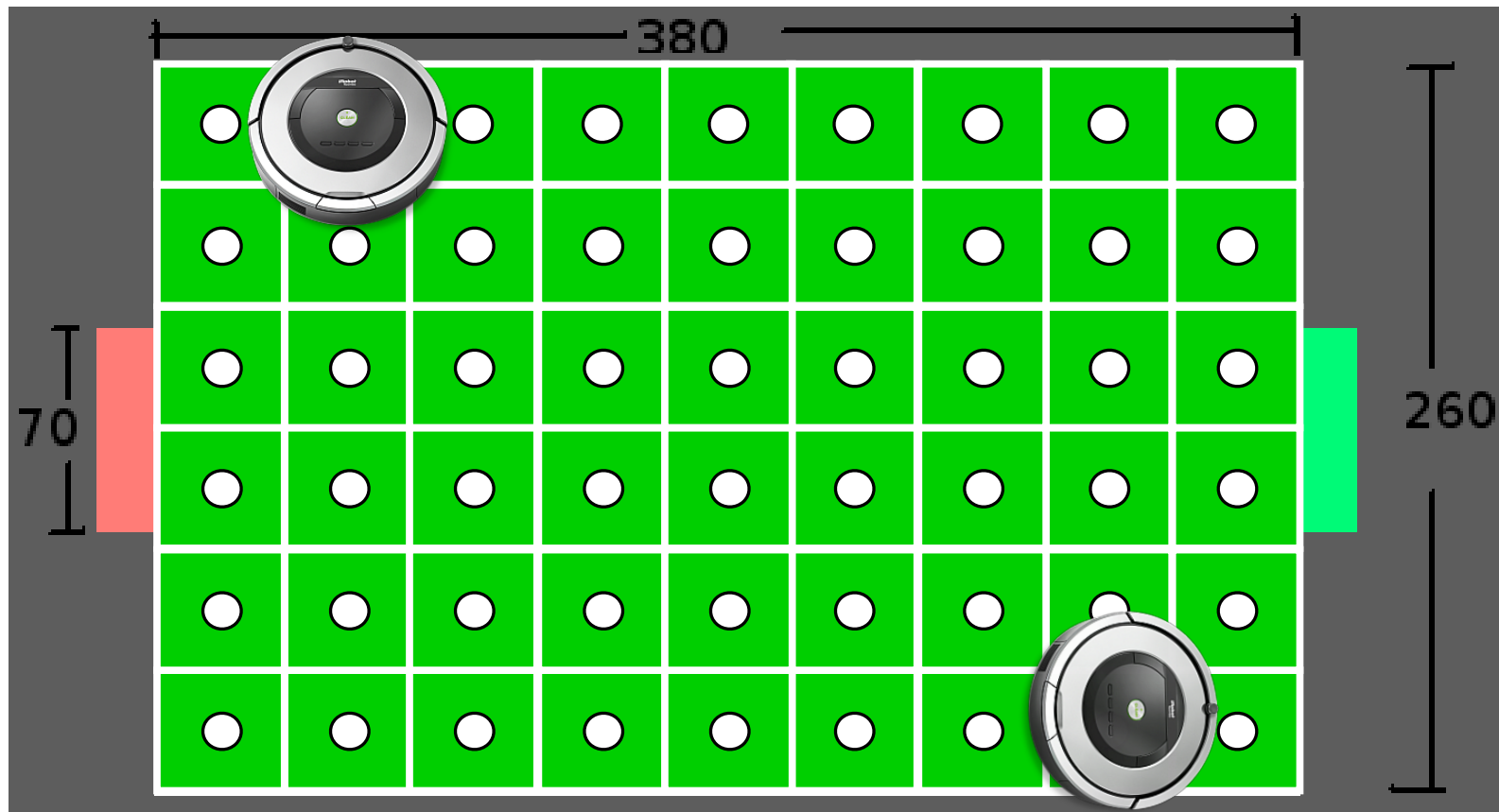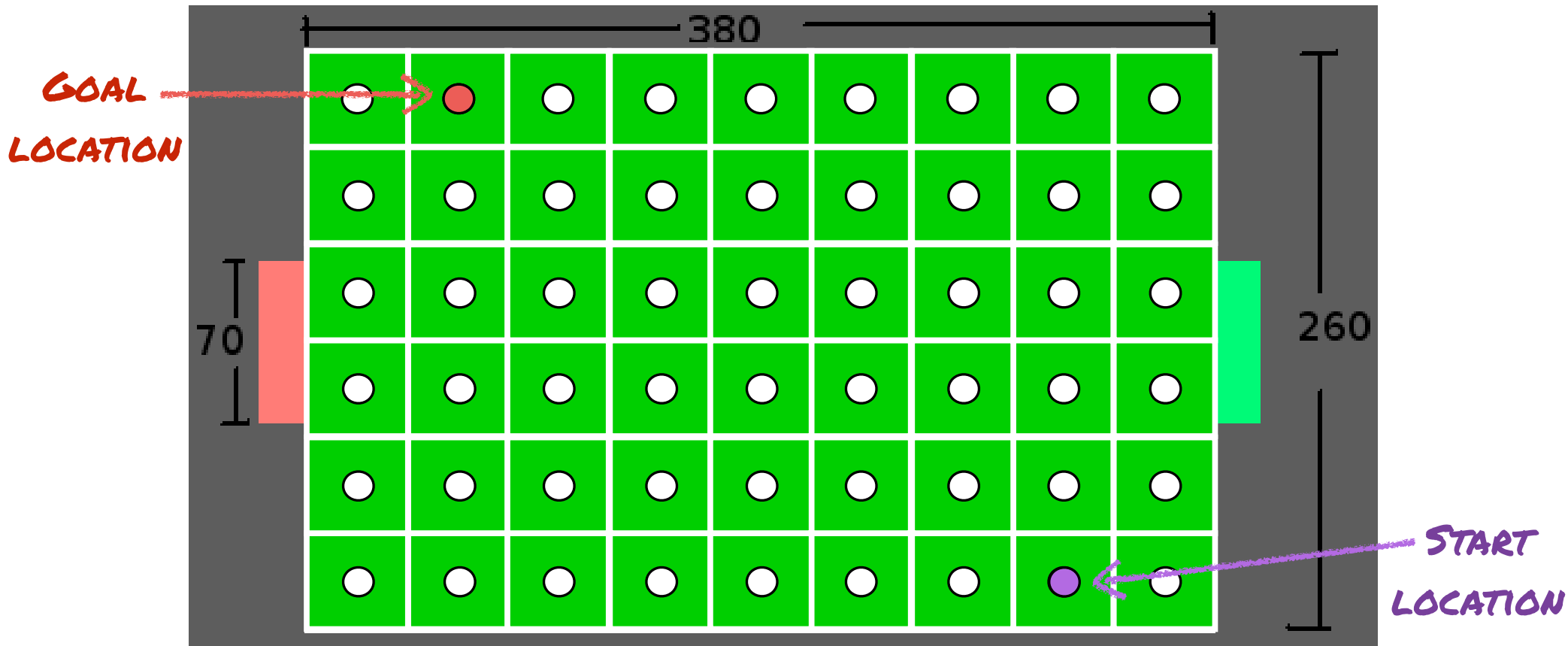# Depth-first search intuition
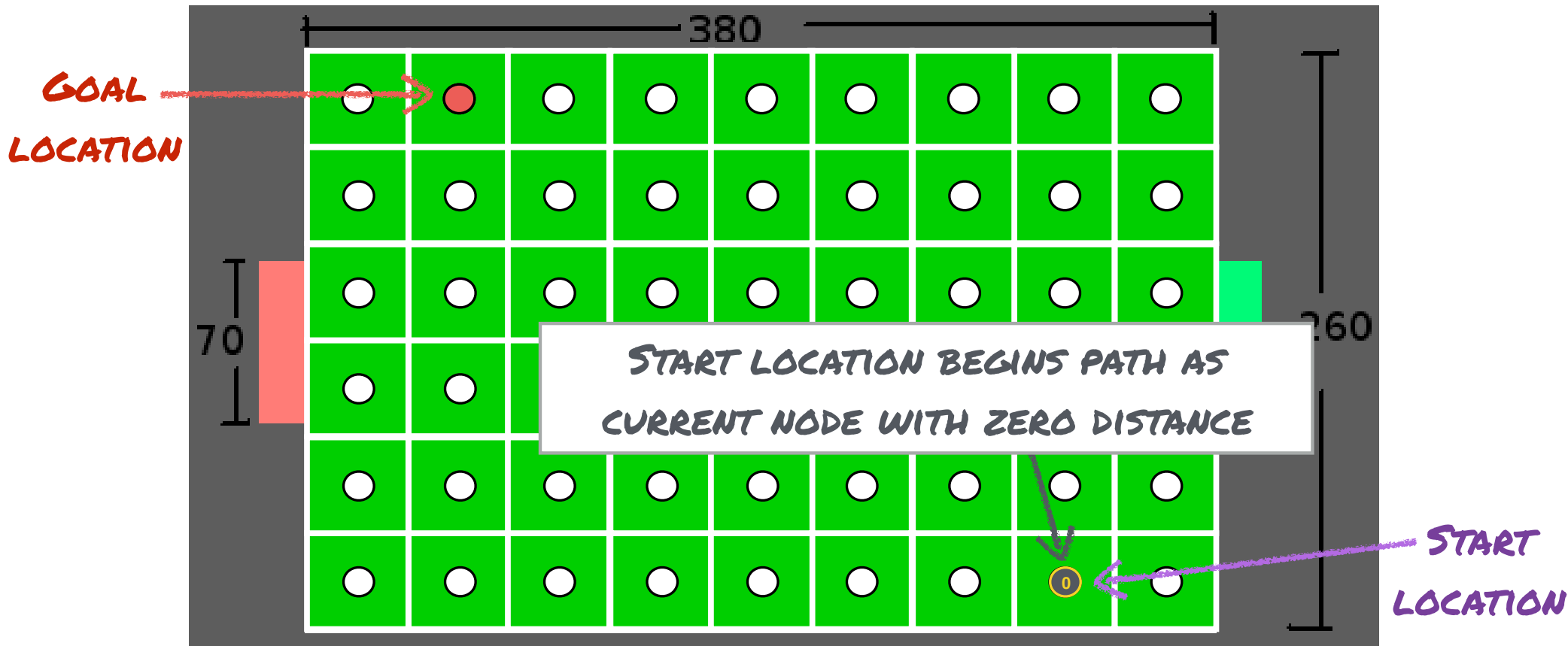
Depth-first search
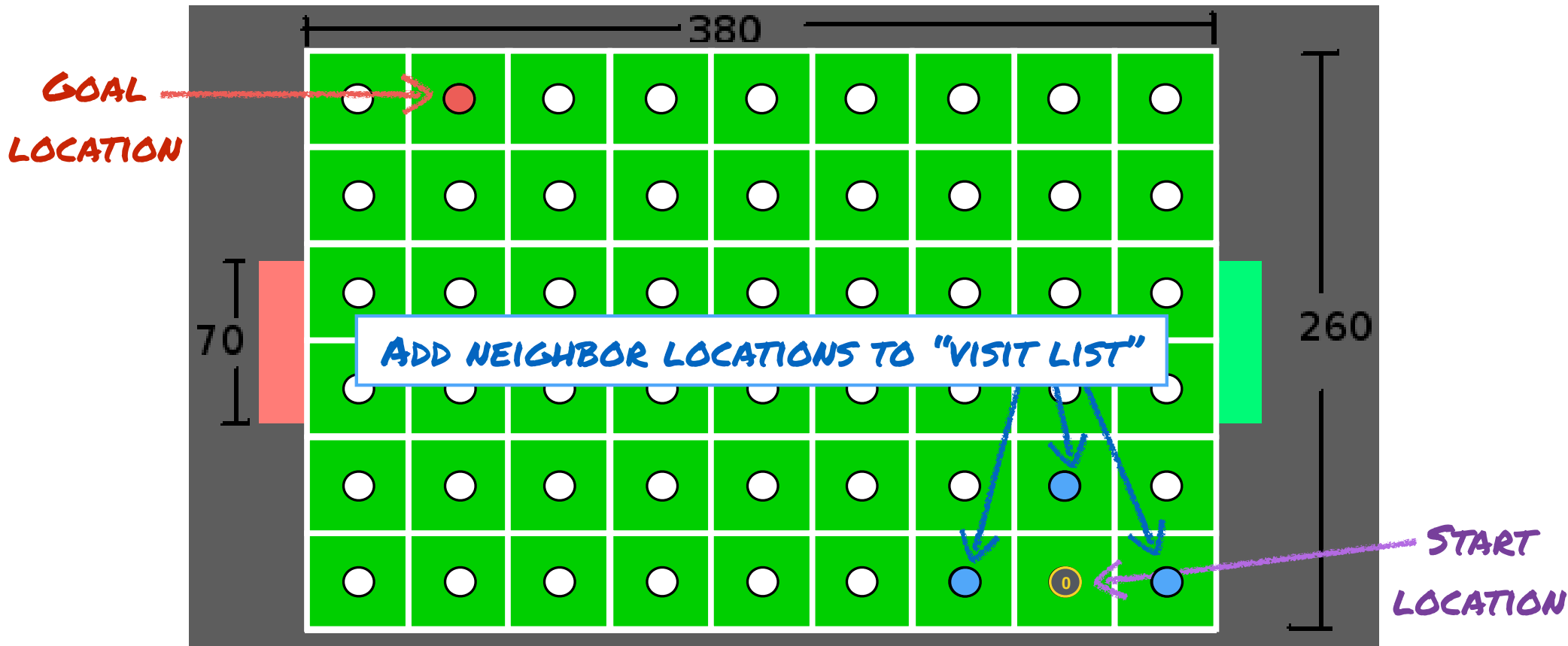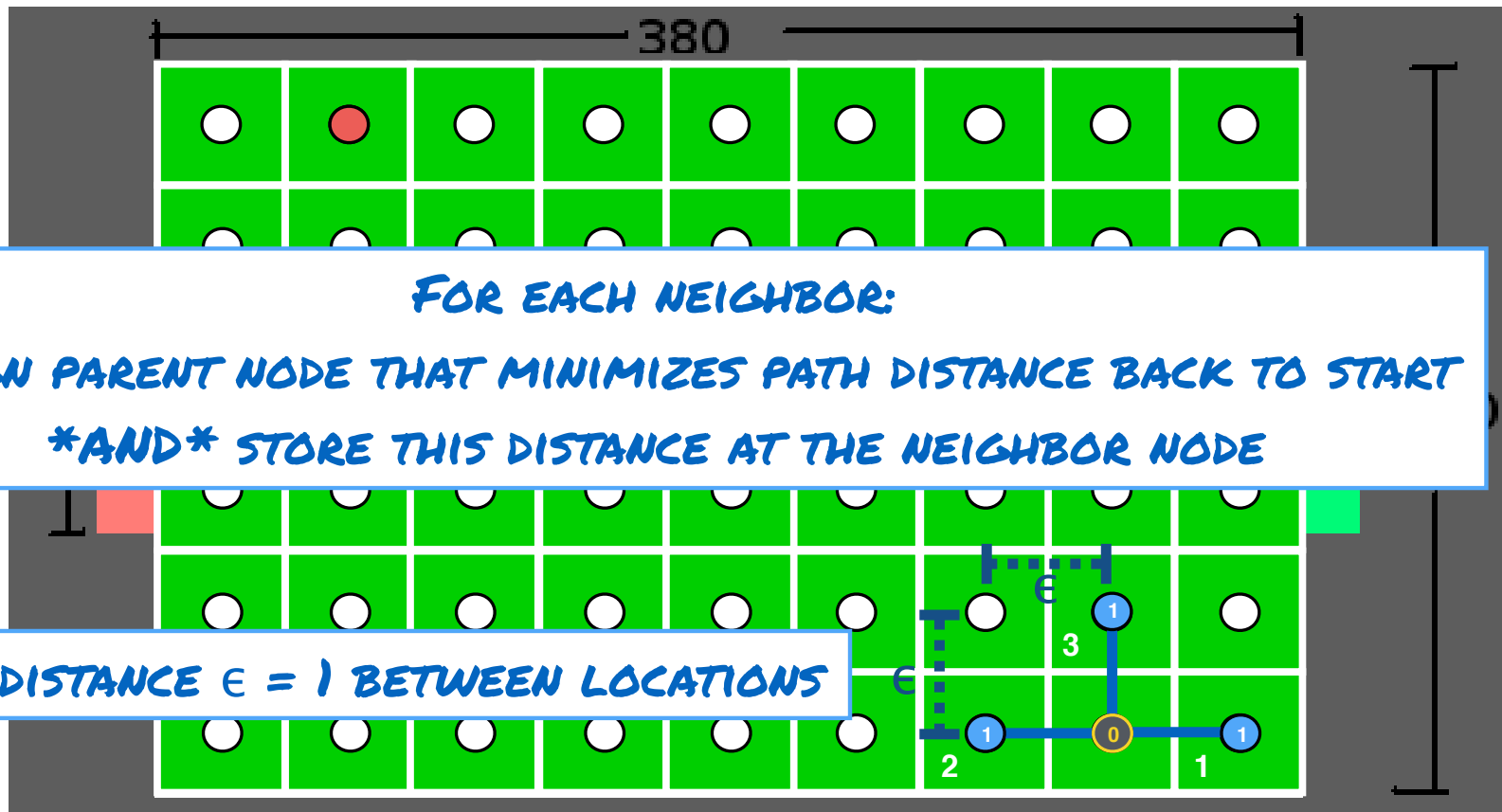
Depth-first search

Depth-first search

Depth-first search

Depth-first search

Depth-first search

For each neighbor:

assign parent node that minimizes path distance back to start *AND* store this distance at the neighbor node

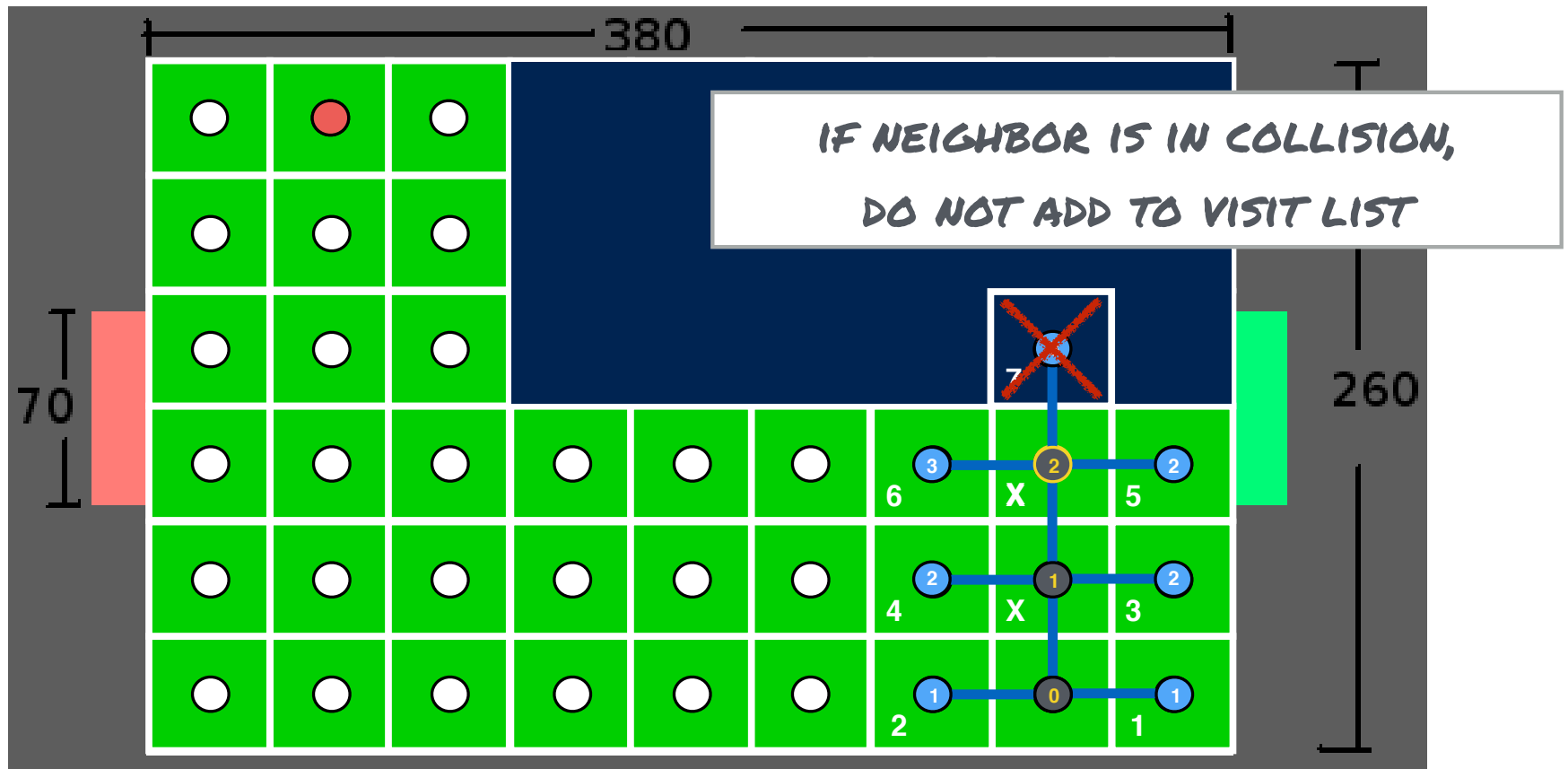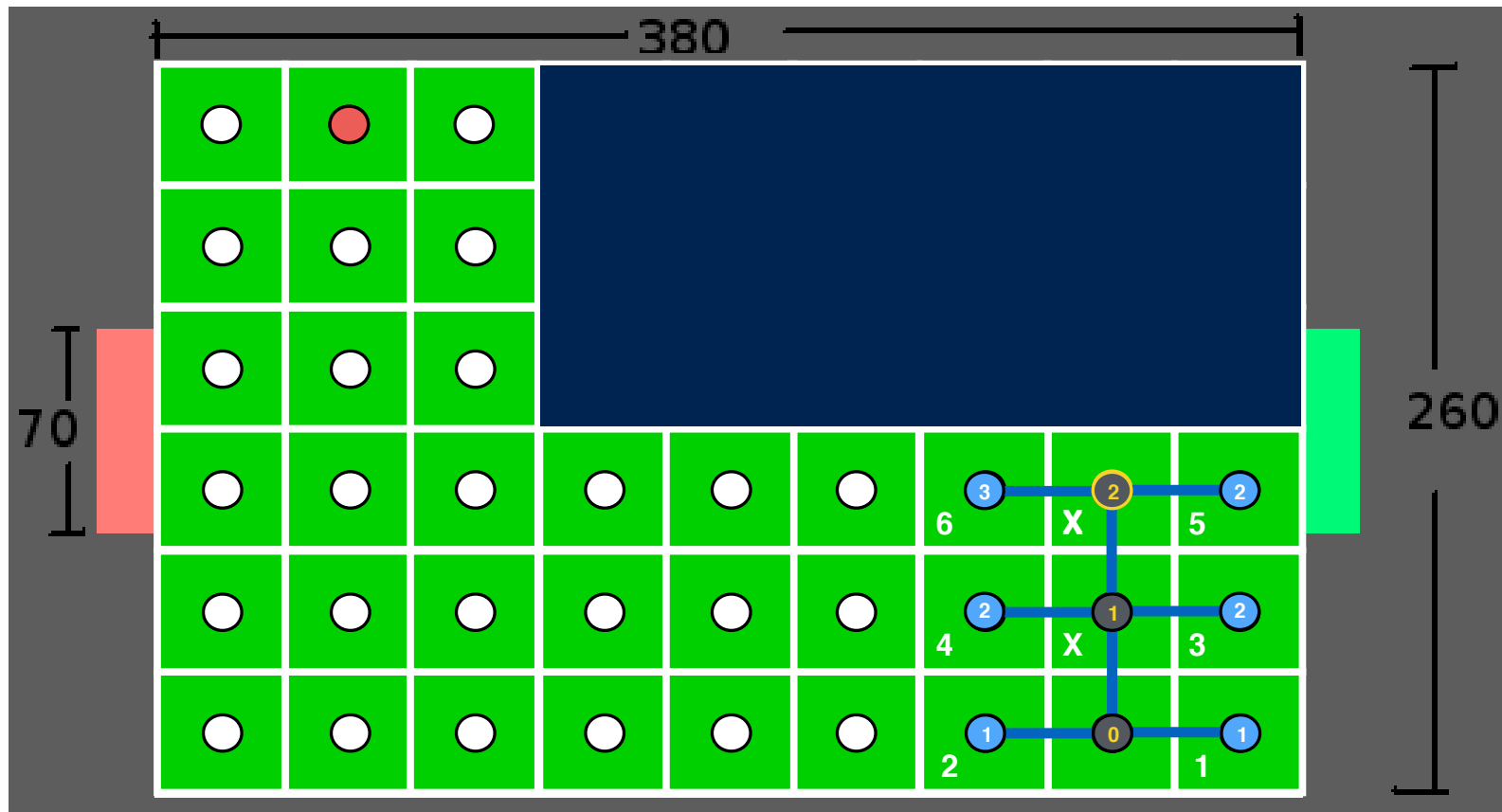# Depth-first search

Depth-first search

Depth-first search

Depth-first search

Depth-first search

Repeat: For each neighbor:

choose parent node that minimizes path distance back to start

AND store this distance (ϵ + ①) at the neighbor node

Depth-first search

Repeat: Visit a neighbor based on order added to visit list and mark as visited

Depth-first search

Depth-first search

IF NEIGHBOR IS IN COLLISION,
DO NOT ADD TO VISIT LIST

Depth-first search

Depth-first search

Depth-first search

DO NOT ADD NEIGHBORS IF THEY ARE IN COLLISION
OR HAVE ALREADY BEEN VISITED OR QUEUED

Depth-first search

Depth-first search

Depth-first search

Depth-first search

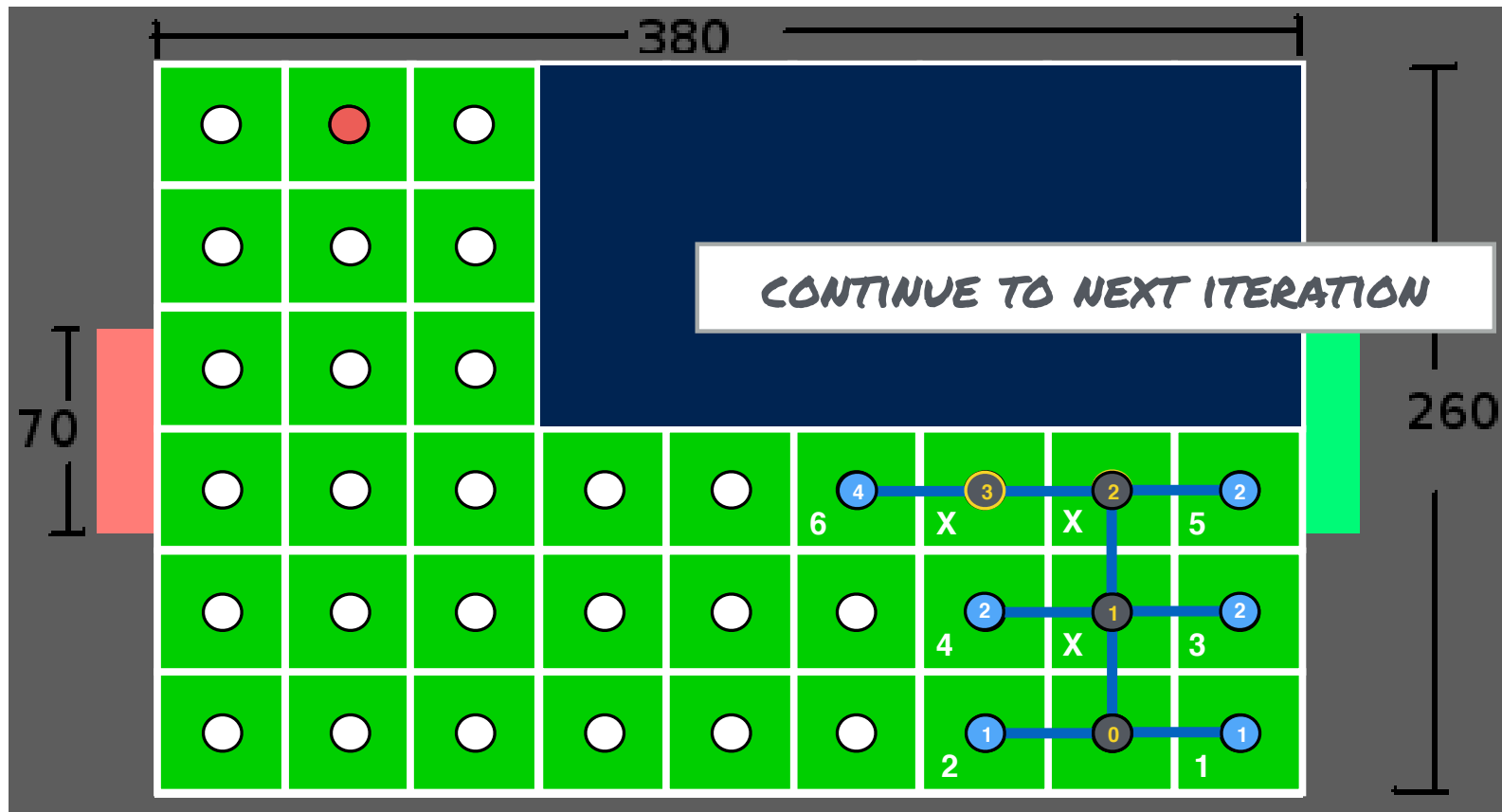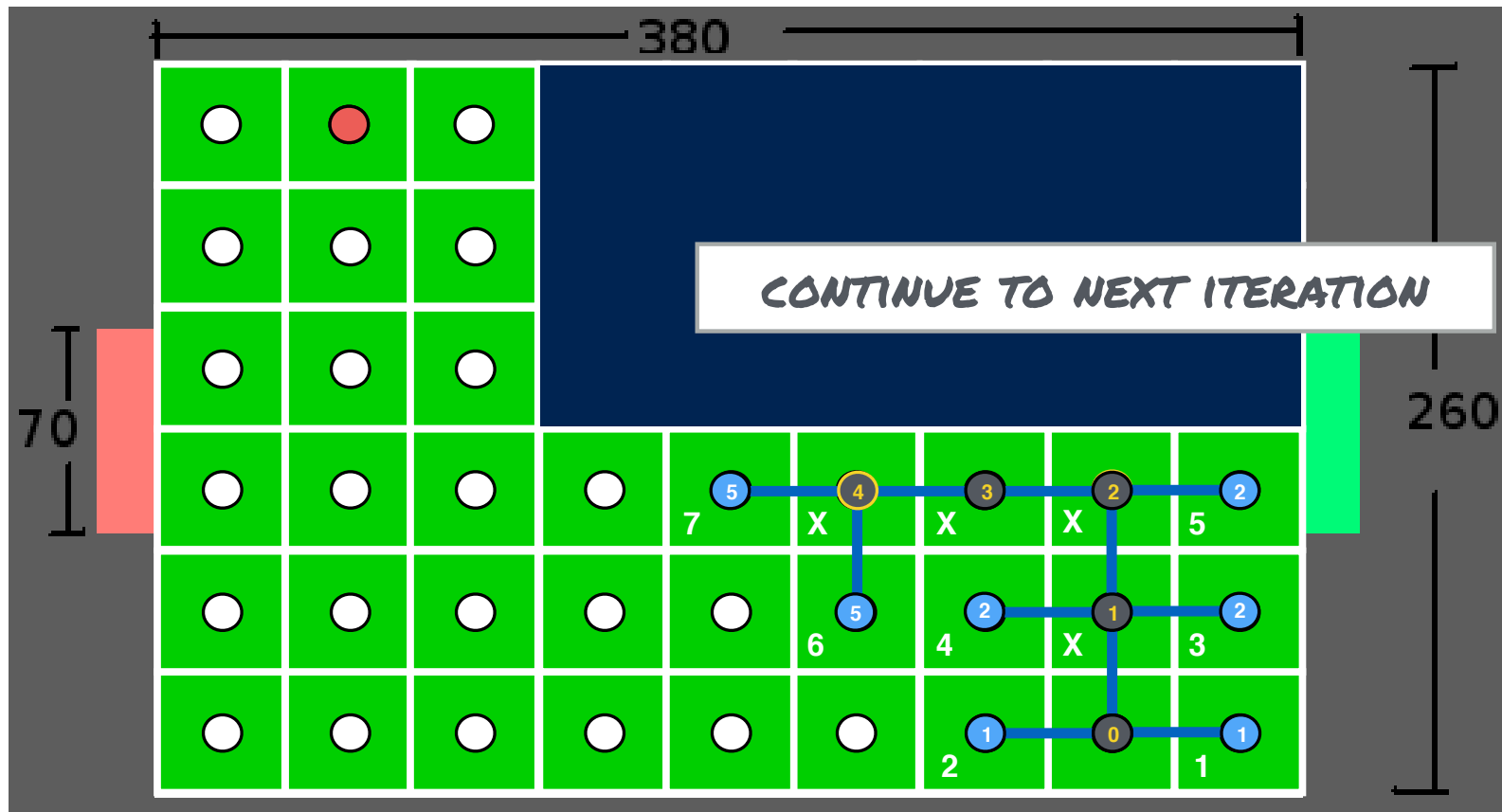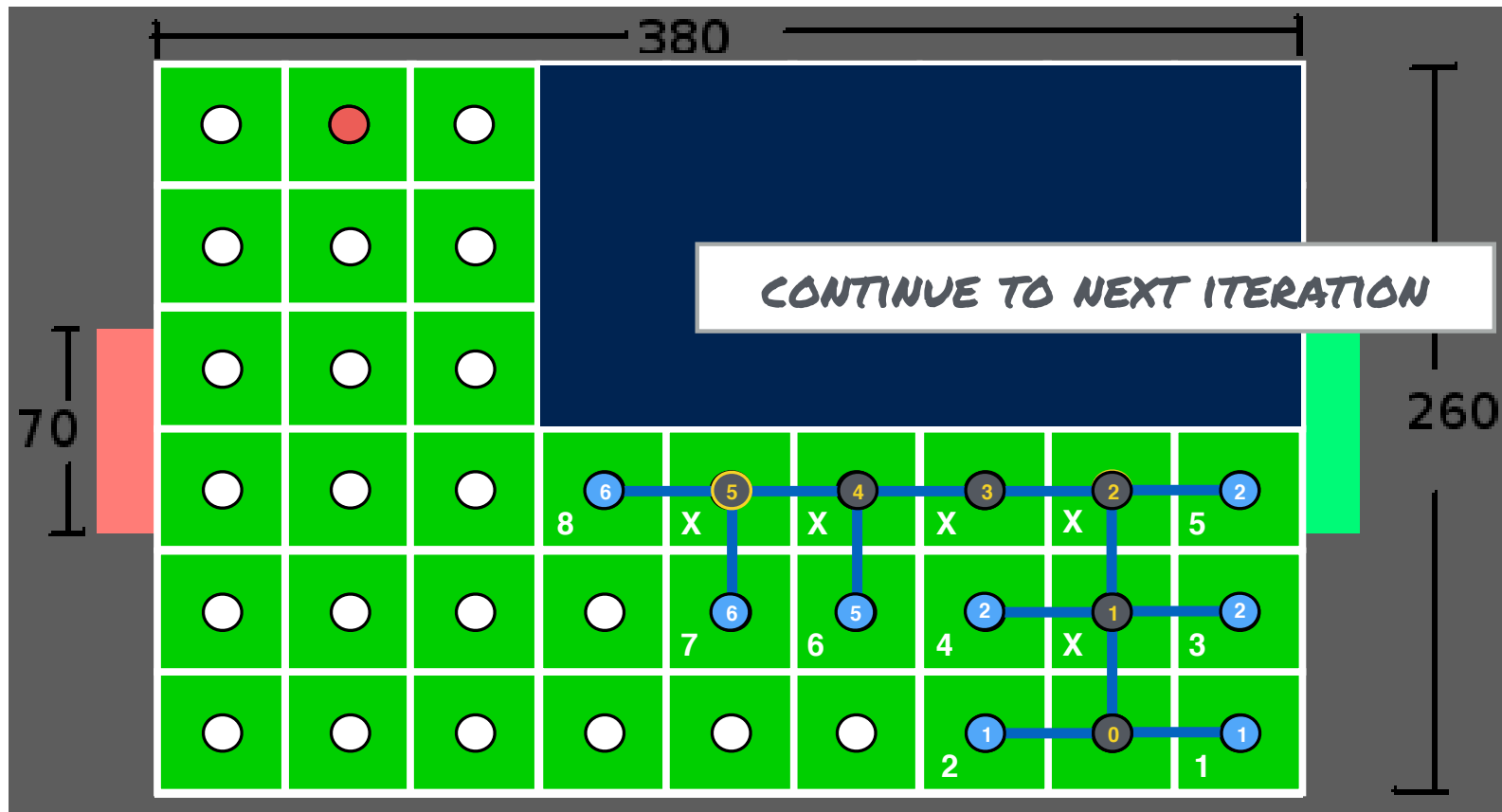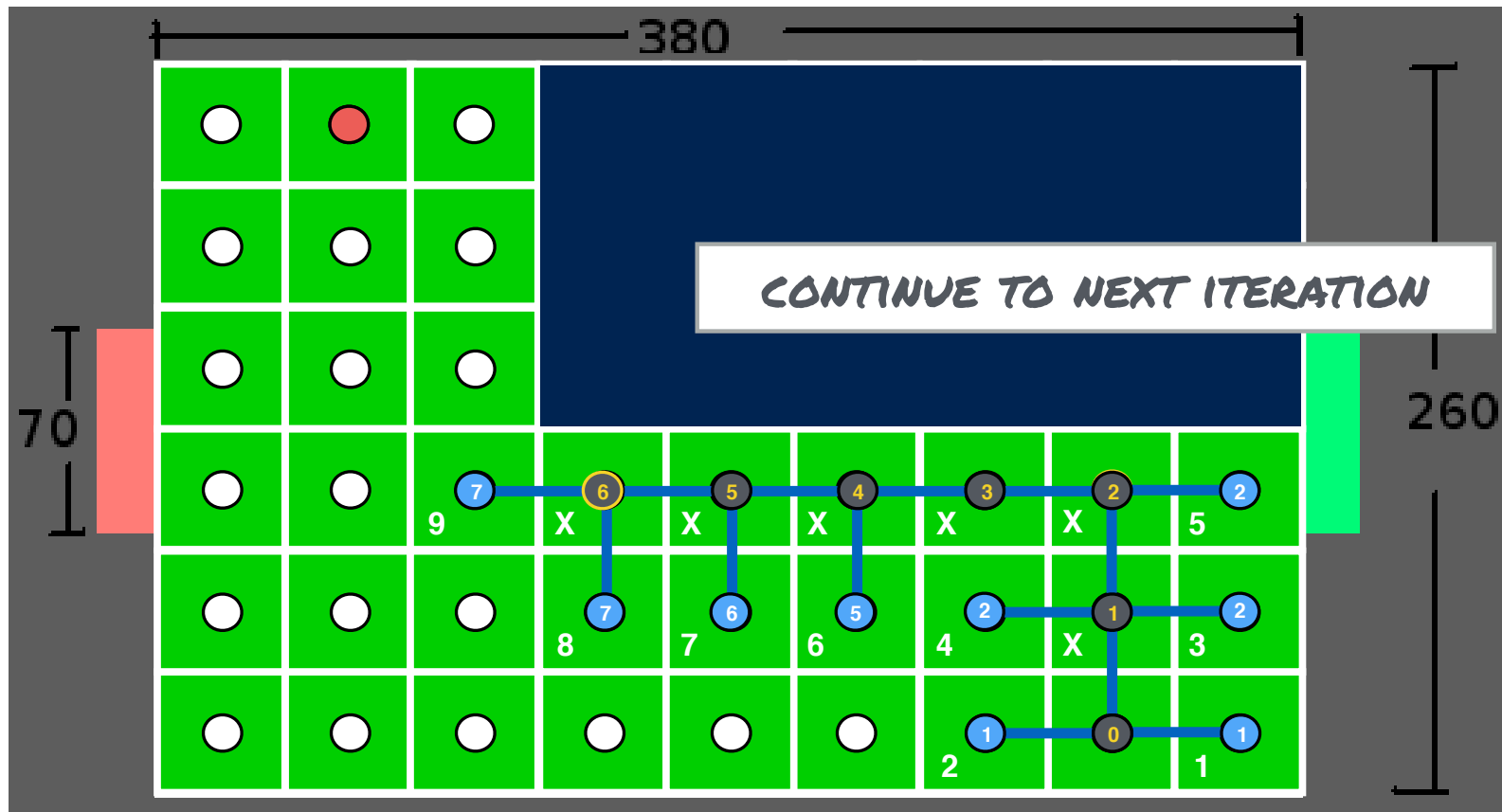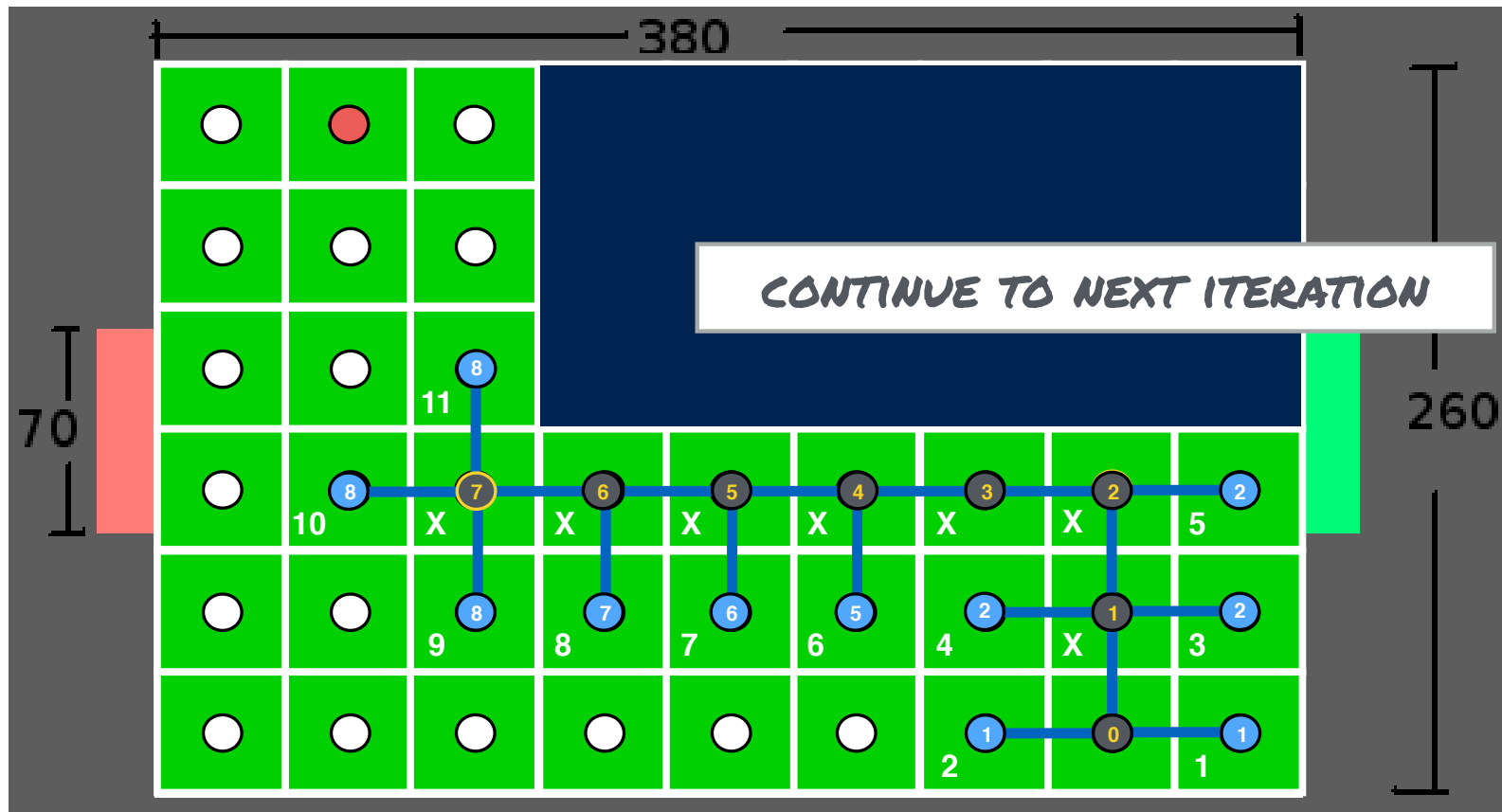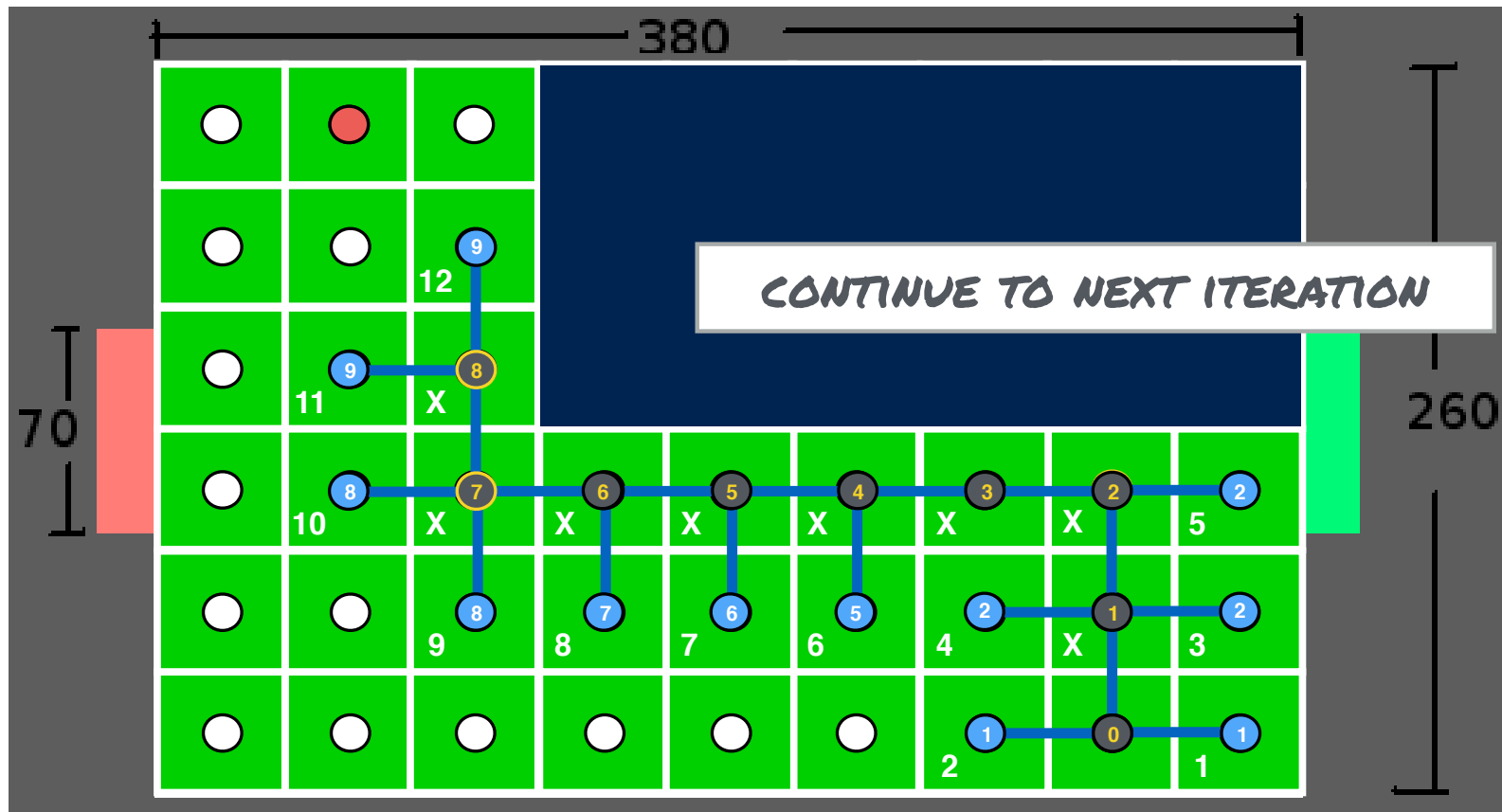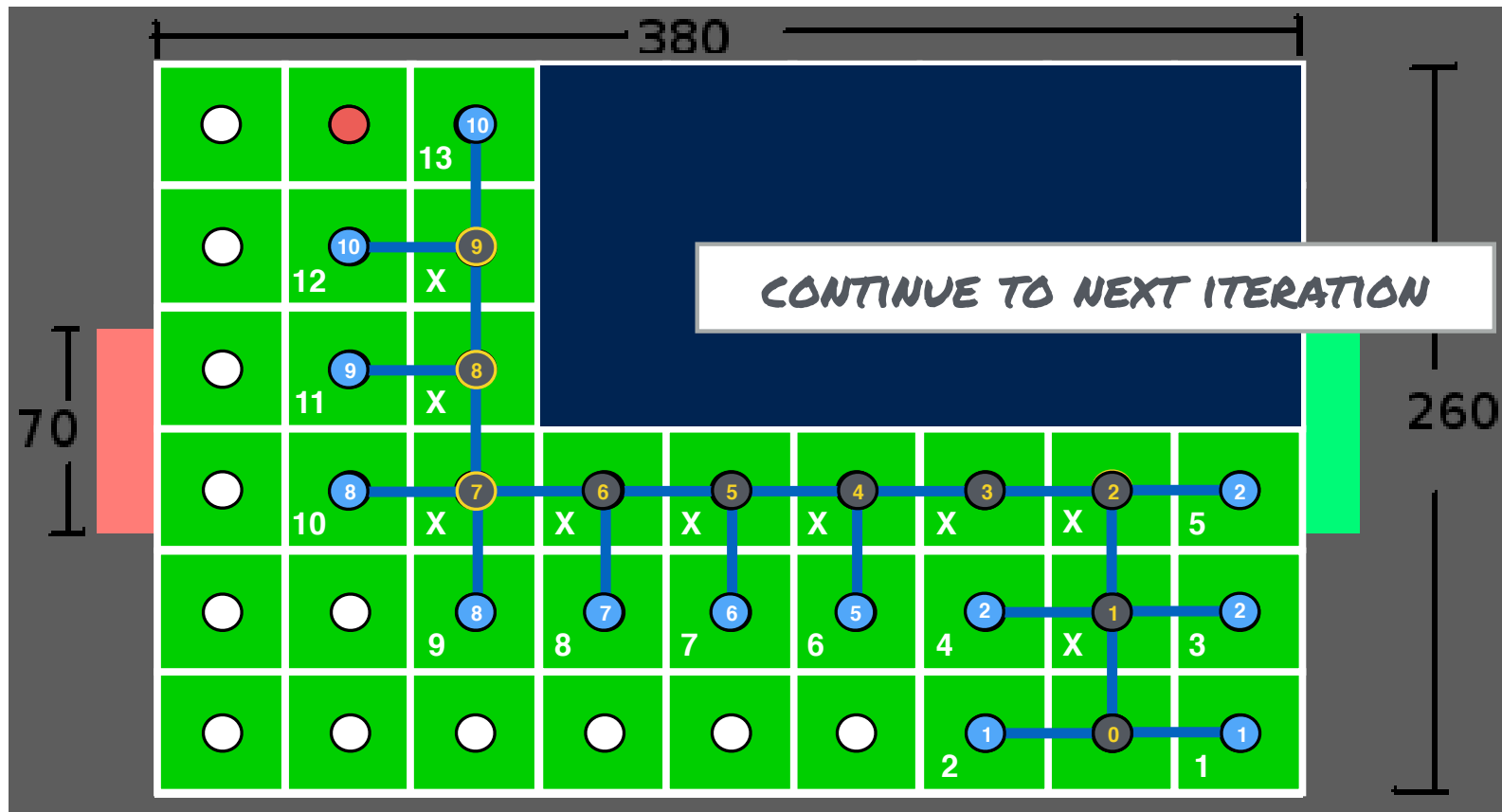# Depth-first search

Depth-first search

Depth-first search

GOAL LOCATION NOW IN VISIT LIST
AND WILL BE VISITED NEXT

Depth-first search

Path found when goal location visited