ALGORITHM FOR ALL SEARCH STRATEGIES

The heart of this algorithm a list of open states. Algorithms also use a list of closed states

```
Put initial state on the open list
while the open list is not empty and
        the first node is not a/the goal

    remove the first node from the open list
    expand it
    put node that was expanded on closed list
    put the generated nodes on the open list *

(if the open list becomes empty without
finding a/the goal, the search fails)
```

* The various search algorithms differ only in how they do this

Breadth-first search (BFS) puts the newly-generated nodes at the end of the open list, behind all nodes already there

Depth-first search (DFS) puts the newly-generated nodes at the front of the open list, ahead of all nodes already there

A* maintains for each node

- A record of the cost of getting there from the initial state (cost-so-far)
- A heuristic estimate of the cost of getting to a/the goal from this state (estimated remaining cost)

A* maintains for each node

- A record of the cost of getting there from the initial state (cost-so-far)
- A heuristic estimate of the cost of getting to a/the goal from this state (estimated remaining cost)

With these two values, it is possible to calculate estimated total cost =
    cost-so-far + estimated remaining cost

A * maintains the open list in increasing order of estimated total cost - i.e. it sorts the newly generated nodes in order of estimated total cost and merges them with the nodes currently on the open list

Search algorithms may use the closed list as follows:

Most simply remove a node from the list of newly generated nodes before adding it to the open list if it is already on the closed list (to avoid exploring the same state twice)

A * retains a node that duplicates one already on the closed list if its estimated total cost is less than the estimated total cost of the one that is closed.

Also, when merging newly generated nodes with the open list, if a state is already on the open list (because it has gotten there by expanding some other node) A* keeps the version with the lowest estimated total cost