

```
In [181... import subprocess
```

```
In [ ]: def runSnake(mode, num_obstacles):
        command = ['python', 'snake_starter.py', f'{mode}', f'{num_obstacles}']
        p = subprocess.run(command, capture_output=True).stdout
        try:
            return int(p.decode()[17:])
        except:
            return runSnake(mode, num_obstacles)
```

```
In [183... modes = ['greedy', 'astar', 'dijkstra']
num_obstacles = 0
trials = 10
```

```
In [184... avg_scores = {}
for mode in modes:
    avg_scores[mode] = 0
    for _ in range(trials):
        avg_scores[mode] += runSnake(mode, num_obstacles)
    avg_scores[mode] = avg_scores[mode]/trials
print(avg_scores)
```

```
{'greedy': 30.2, 'astar': 86.0, 'dijkstra': 81.4}
```

```
In [185... num_obstacles = 10
modes = ['greedy', 'astar', 'dijkstra']
```

```
In [186... avg_scores = {}
for mode in modes:
    avg_scores[mode] = 0
    for _ in range(trials):
        avg_scores[mode] += runSnake(mode, num_obstacles)
    avg_scores[mode] = avg_scores[mode]/trials
print(avg_scores)
```

```
{'greedy': 29.2, 'astar': 78.0, 'dijkstra': 76.3}
```

Greedy is obviously terrible because it doesn't consider the position of obstacles or the snake's body until an obstacle. Astar on average performs 8 points better than dijkstra's both with and without obstacles, but (while I would like to believe astar performs better because I made it myself), I believe that the difference in performance between dijkstra's and astar is statistically insignificant. Assuming both algorithms work properly, astar and dijkstra's will always find the shortest path between the snake head and any given node. This means that for two identical boards, where the obstacles and apples always appear in the same position, dijkstra's and astar should find the exact same path to the next apple, meaning the scores should be identical. Perhaps for some apples, multiple shortest paths exist, which might explain the slight difference in performance. A

few times when I ran the test, dijkstra's performed better than astar, so it's likely that the difference in performance would approach zero as more trials are ran.