Python Chapter 8: Heap Queues

Ezequiel Torres

June 27, 2024

Table of contents

Basics of Heaps

Heaps are a binary tree where the root is always less than the children

We can use this to sort a list, or to have a list where we know the top element is always the smallest

Heap Example

```
import heapq
h = []
heapq.heappush(h, 3)
heapq.heappush(h, 6)
heapq.heappush(h, 2)
heapq.heappush(h, 9)
print(h[0]) # Prints 2
```

Heap Pop Example

```
import heapq
h = []
heapq.heappush(h, 3)
heapq.heappush(h, 6)
heapq.heappush(h, 2)
heapq.heappush(h, 9)
heapq.heappop(h)

print(h[0]) # Prints 3
```

Heapify Example

```
import heapq
h = [3,6,2,9]
heapq.heapify(h)

print(h[0])
```

Heap Sort

```
import heapq
h = []
for value in [3,6,2,9]:
    heapq.heappush(h, value)

# Prints 2,3,6,9
print([heapq.heappop(h) for i in range(len(h))])
```