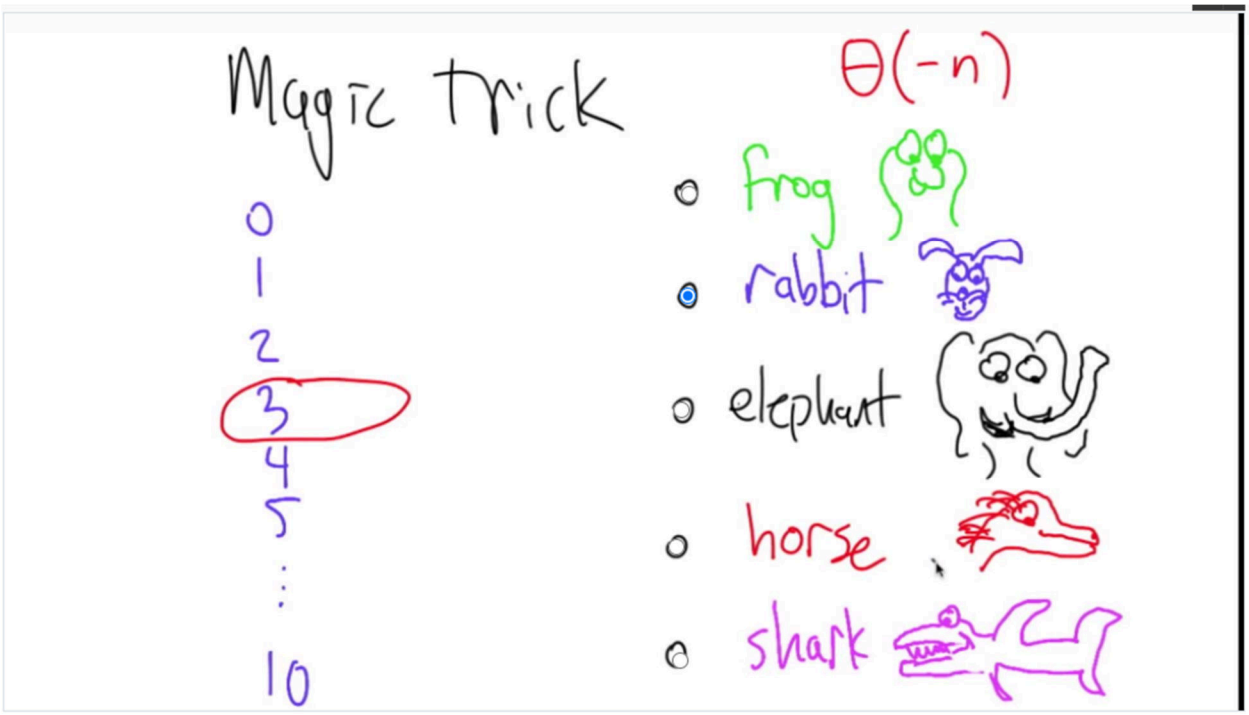
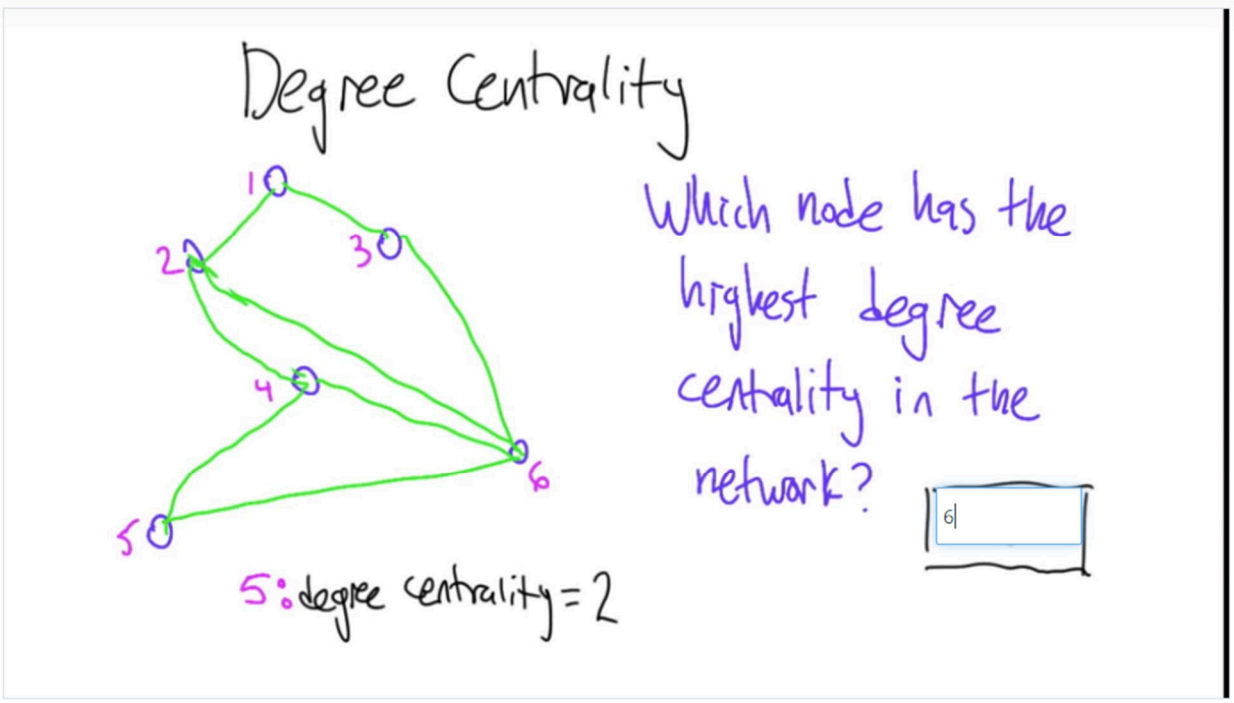
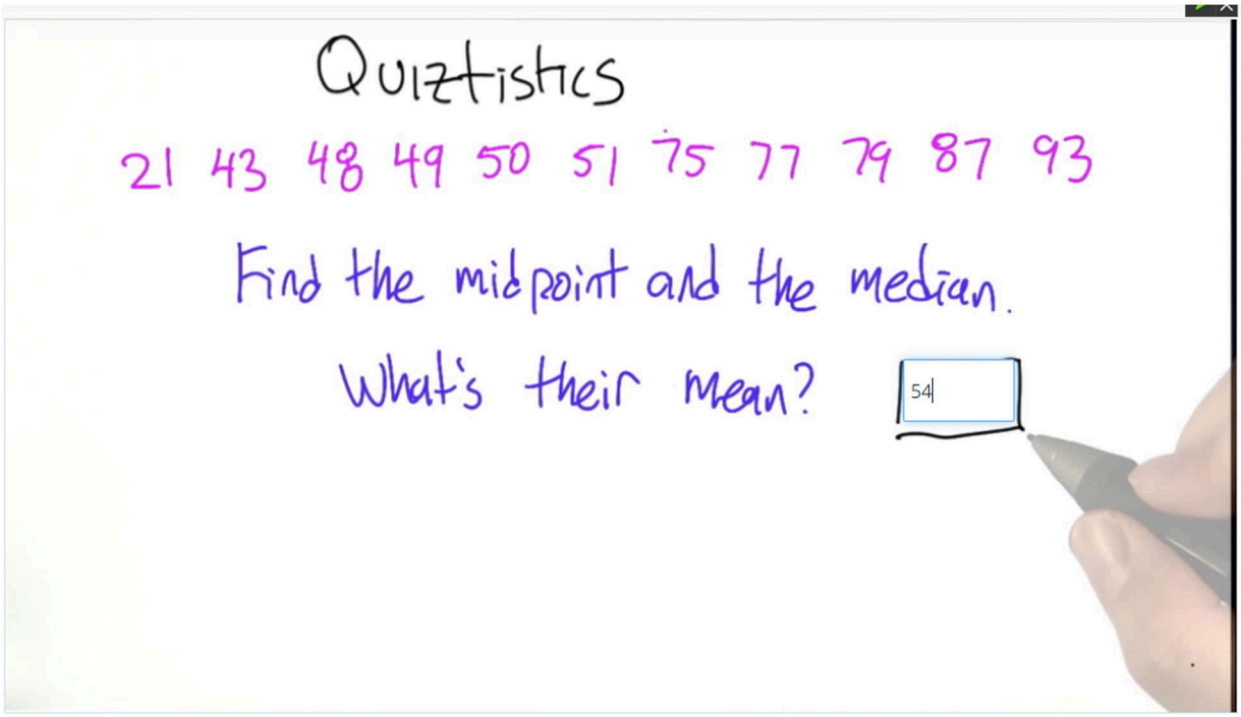
Introduction



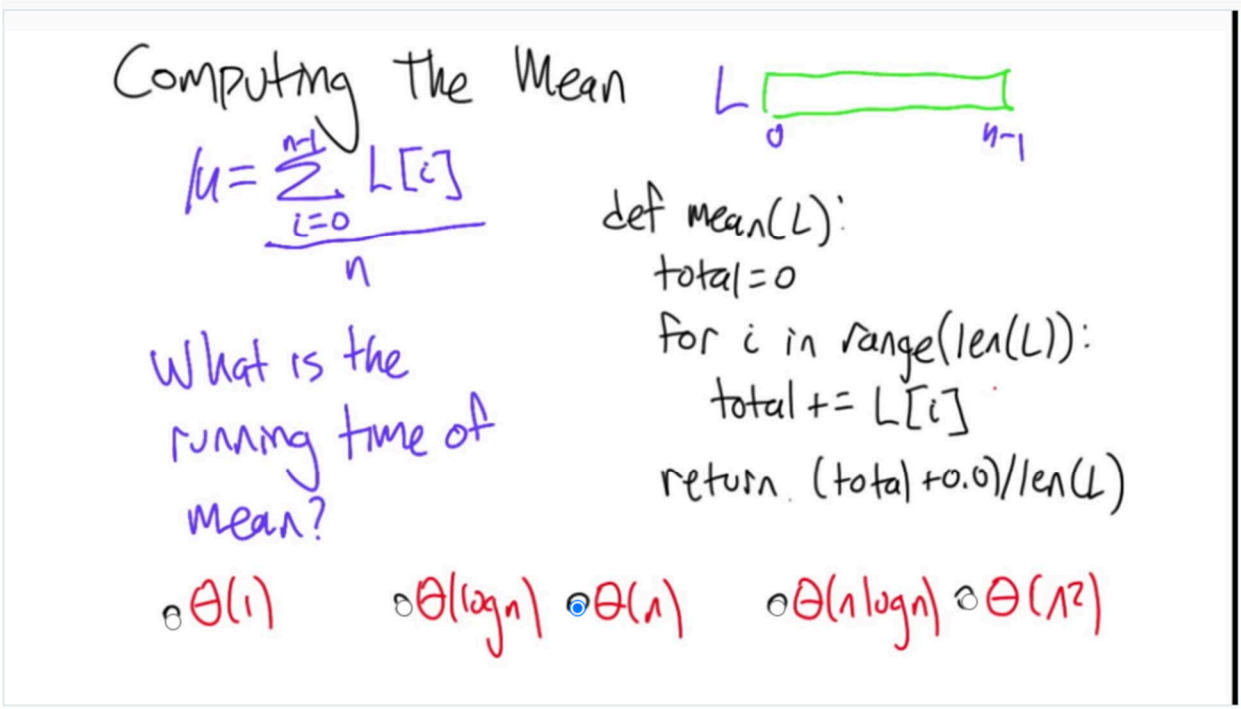
Degree Centrality



Statistics By Sorting



Mean



Extreme Values

```python
#
# Write `max`
#

def max(L):
    max_so_far = L[0]
    for i in range(1,len(L)):
        if L[i] > max_so_far:
            max_so_far = L[i]
    return max_so_far

def test():
    L = [1, 2, 3, 4]
    assert 4 == max(L)
    L = [3, 6, 10, 9, 3]
    assert 10 == max(L)
```

Order Statistics

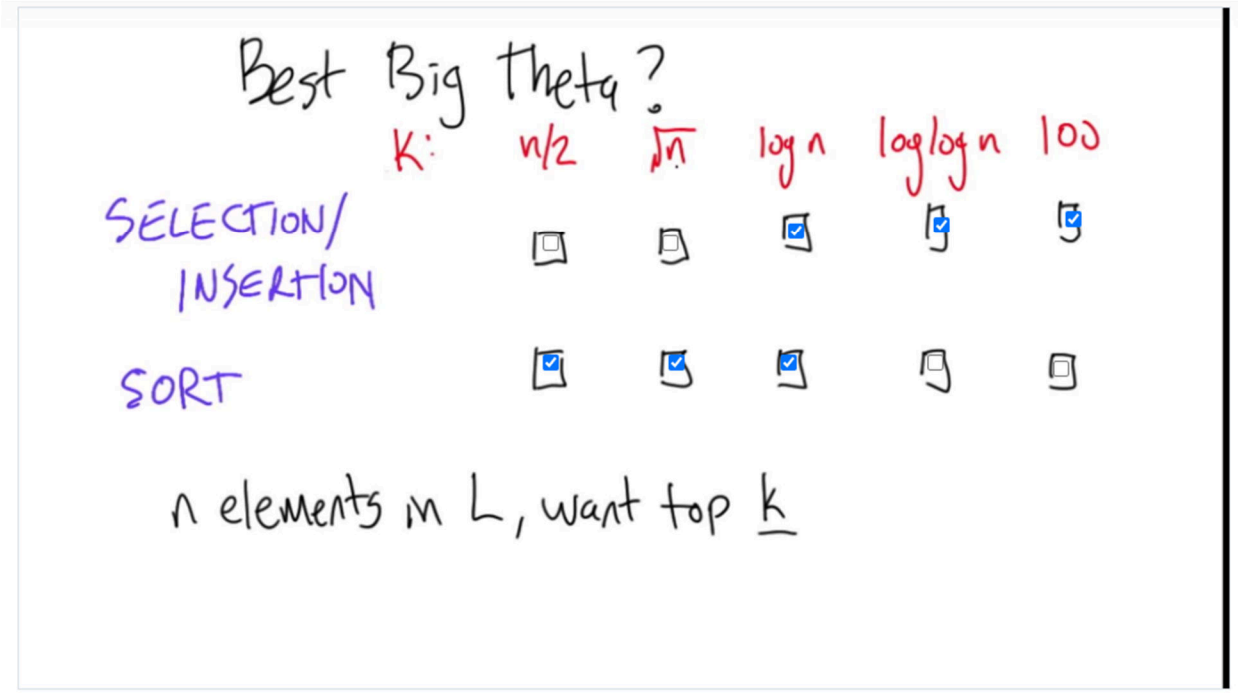
2nd Most Popular Name

yob1995.txt

Write python code to find the 2nd most popular female name given in the US in 1995.

- Michael
- Jessica
- Zuzanna
- Ashley
- Matthew
- Taylor

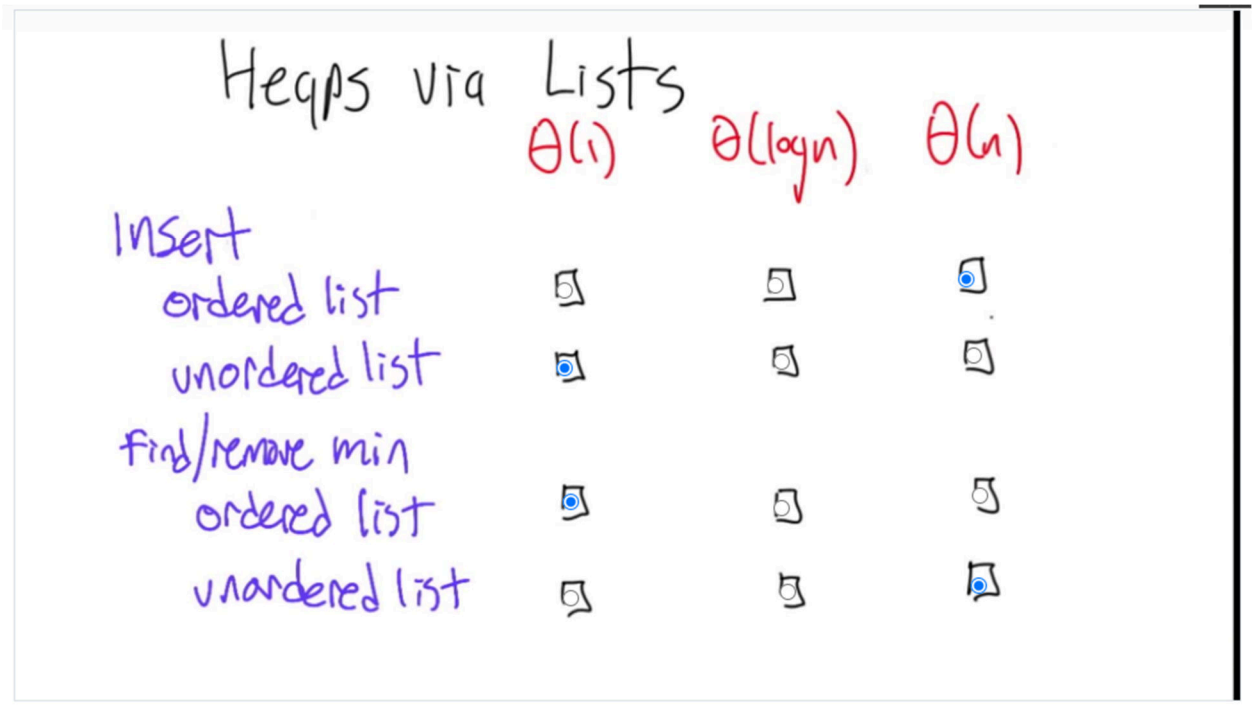
Top K Problem



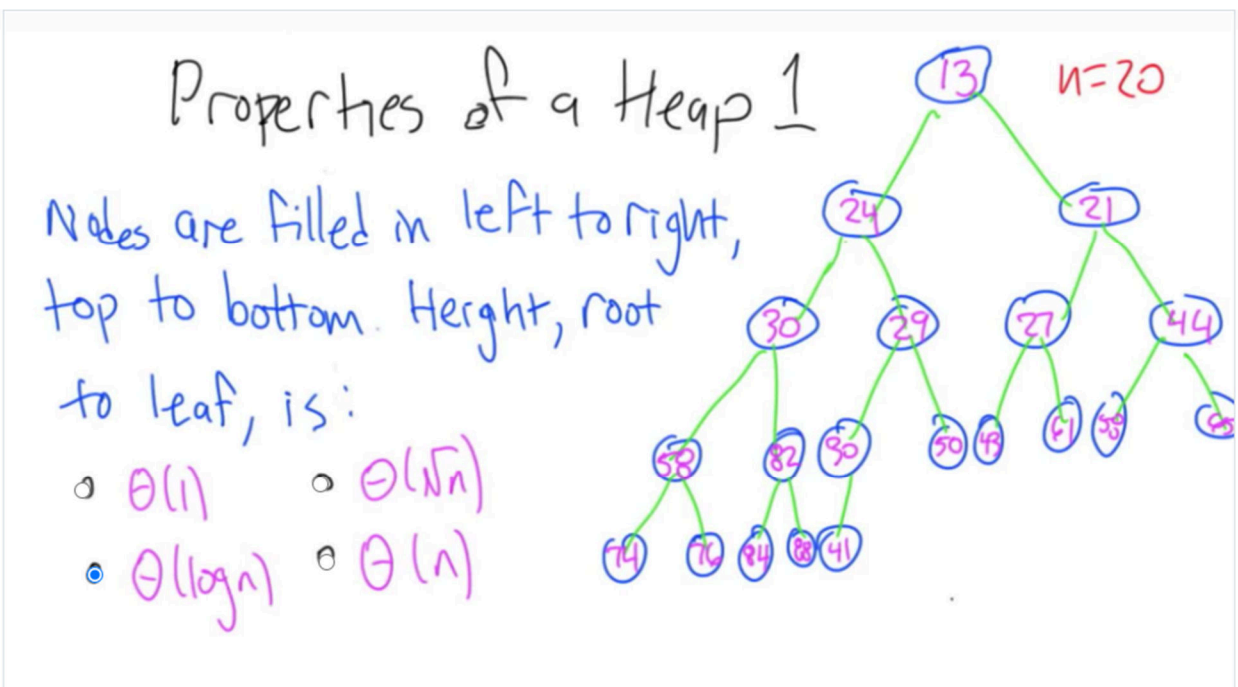
Partitioning Around V

```python
def partition(L, v):
    smaller = []
    bigger = []
    for each in L:
        if each < v:
            smaller.append(each)
        if each > v:
            bigger.append(each)
    # your code here
    return smaller+[v]+bigger
```

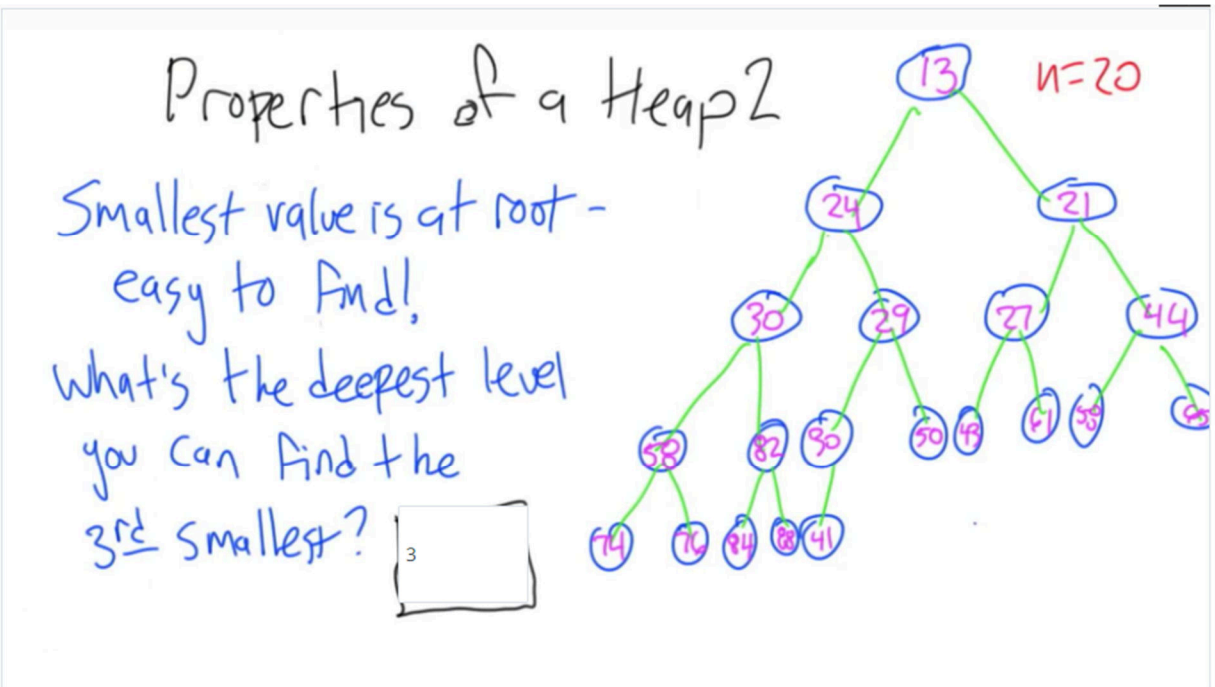
Heaps of Fun



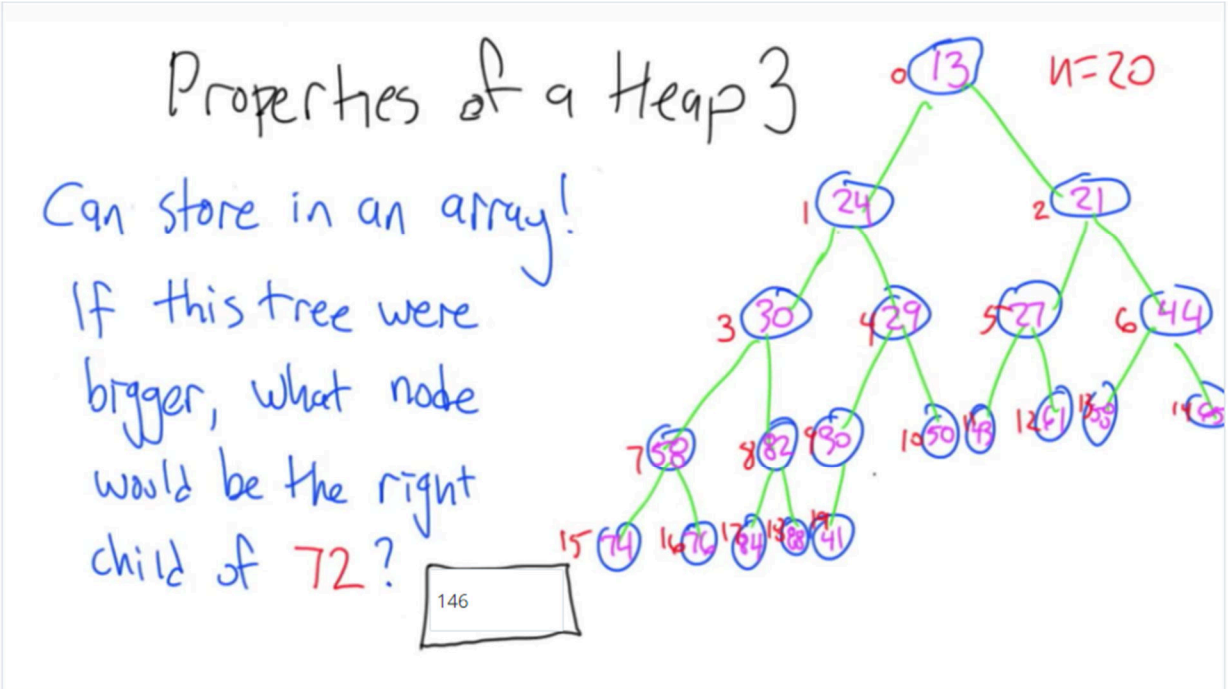
Heap Height



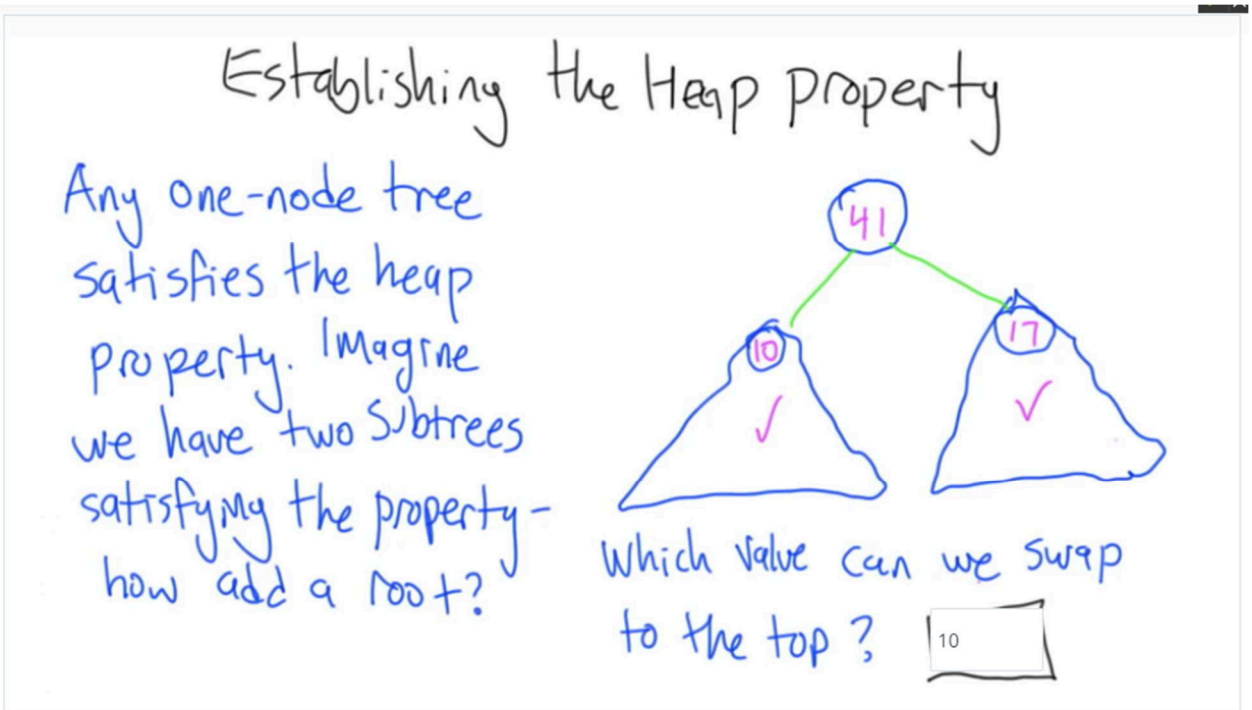
Properties of a Heap



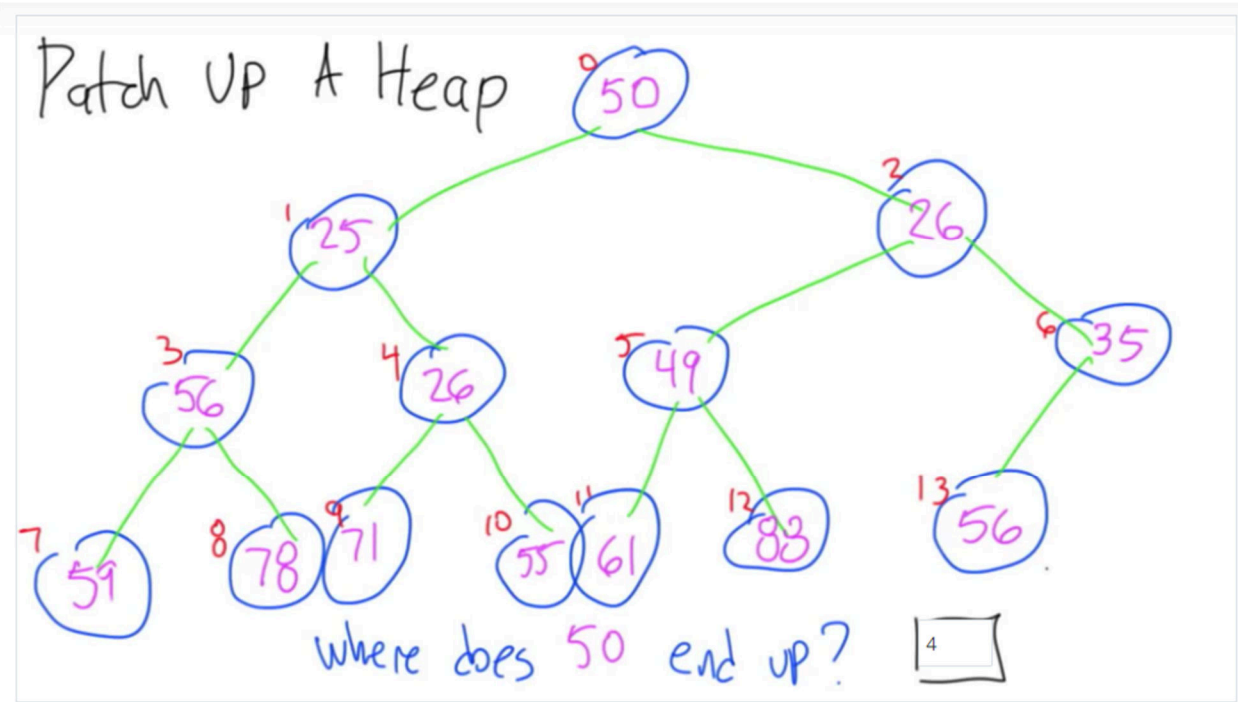
Heap Number



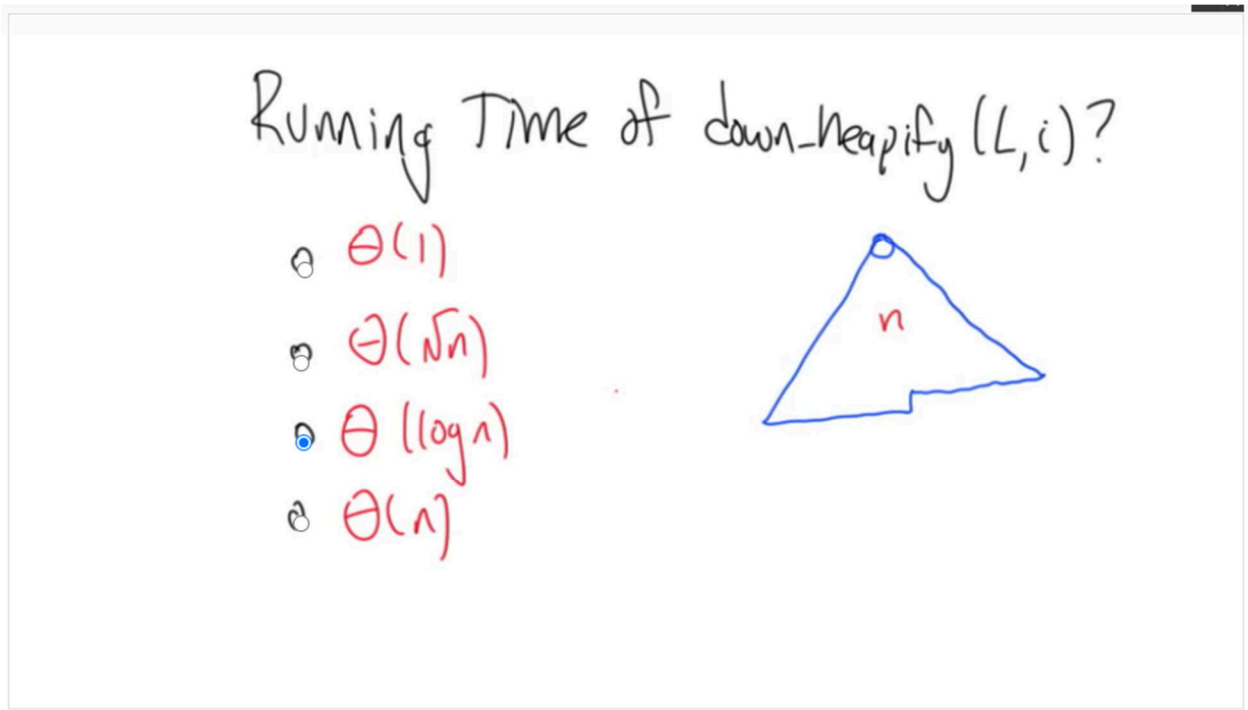
Establishing the Heap Property



Patch Up A Heap



Running Time of Down Heapify



Remove Min

```
def remove_min(L):
    L[0] = L.pop()
    down_heapify(L,0)
    return L
```

Heap Sort Performance

heap sort

```
def heap_sort(L):
    build_heap(L,0)
    while len(L)>0:
        print L[0]
        remove_min(L)
```

running time?

- $\Theta(\log n)$
- $\Theta(n)$
- $\Theta(n \log n)$
- $\Theta(n^2)$