Whiteboard

Monday, March 30, 2020 10:12 AM

Common Search Problems:

- Is an item contained in a list? ← return bool
- Where is the item in the list? ← return int (-1 if not found)
- Max/Min item
- Items in a range
- Items w/ specific type
- How many times a value appears in a list

Alg Analysis: quantify performance

What do we mean by performance?

Interpretation #1: seconds to compute
→ depends on hardware

Interpretation #2: steps the algorithm takes ← want to know the worst case
→ hardware independent ← as a function of the size of the input
→ theoretical measure

EX Counting steps for linear search

L = [23, 77, -34, 45, 99, -4]

Alg #1

```
for i in range(len(L)):
    if L[i] == X:
```

Alg #2

```
found = False
for i in range(len(L)):
```

```
for i in range(len(L)):
    if L[i]==X:
        return True
return False
```

```
found = False
for i in range(len(L)):
    if L[i]==X:
        found = True
return found
```

How many steps does each alg take to find the following values?

Alg #1:

99 : 5
23 : 1
80 : 6

Alg #2

99 : 6
23 : 6
80 : 6

In general:

Alg #1

Ave #steps = $\frac{N+1}{2}$
Min #steps = 1
Worst case = N

Alg #2

where N is # elements in our list

Ave #steps = N
Min steps = N
Max steps (worst case) = N

We are interested in the worst-case performance
→ Upper bound on running time.

"Big-Oh" : notation we use to describe worst-case running time

EX Linear Search is $O(N)$, where N is the length of the list, because in the worst-case we need to check every element in the list