

Data structures and algorithms

Nicky Chorley

July 2016

- These are computer science fundamentals.
- “But computer science is too mathsy!”
- Do we really need all that maths?
- Performance is important: systems are processing more data and we need to serve our users quickly!

What do we mean by performance?

- How long it takes to do something.
- Usually more interested in how time taken depends on size of input.
- Several cases: best case, average case, **worst case**.

What is a data structure?

- Some way of storing a collection of items.
- Common operations: insert an item, remove an item, find a particular item.
- Examples: array, linked list, hash table, tree.

	0	1	2	3
names =	Denise	Oliver	Priya	Kriszta

- In Python:

```
>>> names = ["Denise", "Oliver", "Priya", "Kriszta"]  
>>>
```

Data structures: arrays

names =

0	1	2	3
Denise	Oliver	Priya	Kriszta

```
>>> names[1]
'Oliver'
>>>
```

Data structures: arrays

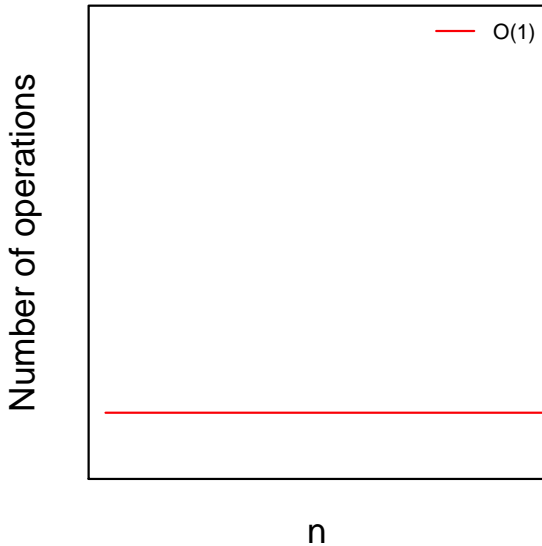
names =

0	1	2	3
Denise	Oliver	Priya	Kriszta

```
>>> names[3]
'Kriszta'
>>>
```

- Time taken to access an item = c
- We call this **constant time** and write $O(1)$.

Constant time



Data structures: linked lists



Data structures: linked lists



Data structures: linked lists

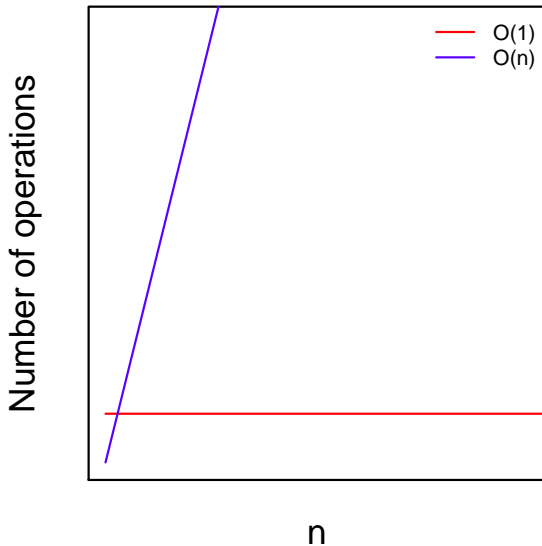


Data structures: linked lists



- Time taken to access an item = $c \times n$
- We call this **linear time** and write $O(n)$.

Linear time



What is an algorithm?

- A sequence of steps for solving a problem.
- Examples: searching, sorting.



Algorithms: searching

- Finding your name badge at Codebar.
- Two ways of solving this problem!

Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

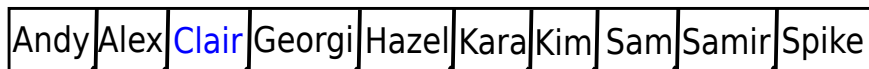
Algorithms: sequential search

Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

Algorithms: sequential search

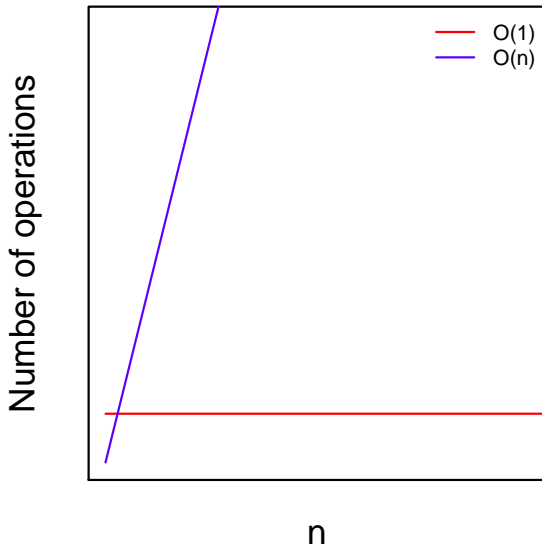
Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

Algorithms: sequential search



- Time taken =
time to access 1 item \times number of items accessed
- The worst case is then
Time taken = $c \times n$.
- We call this **linear time** and write $O(n)$.

Linear time



Algorithms: binary search

- Step: 1

Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

Algorithms: binary search

Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

Algorithms: binary search

- Step: 2

Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

Algorithms: binary search

- Step: 3

Andy	Alex	Clair	Georgi	Hazel	Kara	Kim	Sam	Samir	Spike
------	------	-------	--------	-------	------	-----	-----	-------	-------

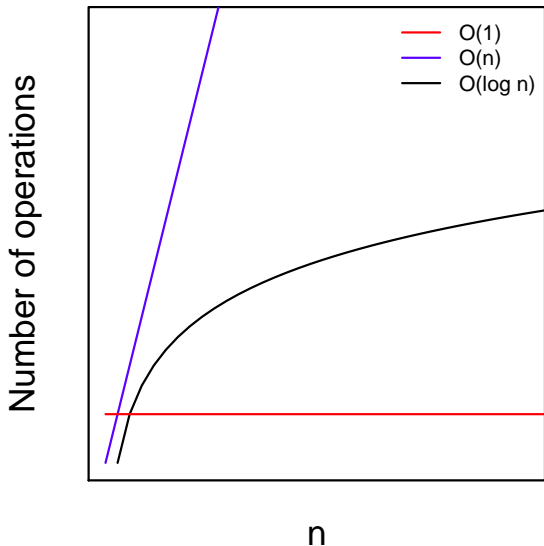
How long does binary search take?

- Time taken =
time to access 1 item \times number of items accessed
- After 1 accessing 1 item, we have $\frac{n}{2}$ items remaining.
- After 2 accessing items, we have $\frac{n}{4}$ items remaining.
- ...

How long does binary search take? (2)

- The worst case is when we can't halve any more, which ends up as
Time taken = $c \times \log_2 n$
- We call this **logarithmic time** and write $O(\log_2 n)$ (or $O(\log n)$ or $O(\lg n)$).

Logarithmic time



Summary

- Data structures are different ways to store items - know your data to make a choice!
- Algorithms are “recipes” for solving problems - there may be several that solve the same problem.

Notation	Name	Example
$O(1)$	Constant	Array element access
$O(\log n)$	Logarithmic	Binary search
$O(n)$	Linear	Sequential search

- Grokking Algorithms, by Aditya Bhargava, Manning:
<https://www.manning.com/books/grokking-algorithms>
- Computer science fundamentals for self-taught programmers, Justin Abrahms, PyCon 2014:
<https://www.youtube.com/watch?v=nEquiifH33w>
- Me!
 - Slides at <http://nicky.xyphias.com/talks>
 - Slack: ndchorley
 - Email: ndchorley@gmail.com