CS1600
12/2 - Tues.

Admin

- Quiz 4 Fri
- Quiz review 4-7pm today Ryder 266
- Hw7 due 9pm
- 2nd chance Hw due 12/5 9pm
  - ↳ no grace period
  - ↳ pls submit early :)
- Final exam 12/12 8-10am
  12/13 1-3pm

Agenda

1. Algorithmic runtime
2. Complexity classes

1. Algorithmic Run-Time

↳ How long will my code take to run?

▶ 00:00

Context: when / how much we've cared

(ex) 1975
Proc: 14MHz
RAM: 16KB
we had to care!

1995
Proc: 144MHz
RAM: 2MB
we cared less

2020
Proc: 1.4GHz
RAM: 6GB

Factors:
- loops
- length of code
- memory allocation
- memory size *
- processor speed *
- compiled / interpreted
- prog language
- import / libraries

the difference now: data!

websites: 2000 (72mil) 2018 2.6bil

tweets: 2010 5mil 2024 500mil

self-driving car: 4 terabytes AI training: 1 petabyte

Algorithmic Run-Time

Ignore: prog lang, mem size, processor

Care: design of your solution

Pseudocode

(A)

- input - list, sequence, set
- input size - n

↳ How many steps on input of size n?

↳ as n grows arbitrarily large

ex
```
sum = 0          1 step
for i = 1 to n   n steps
    sum = sum + A[i]   n steps
```

Run-time $T(n) = 1 + n + n = 2n + 1$

2. Complexity Classes

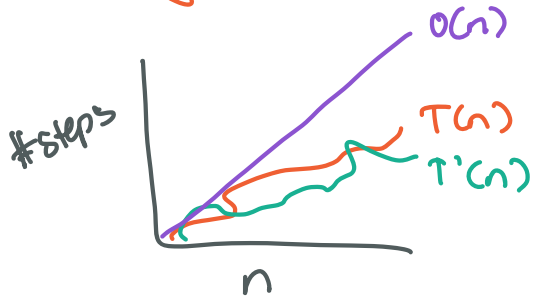
↳ compare algorithms run-times

- put similar run-times in same bucket
- same upper bound - big O

ex $T(n) = 2n+1$

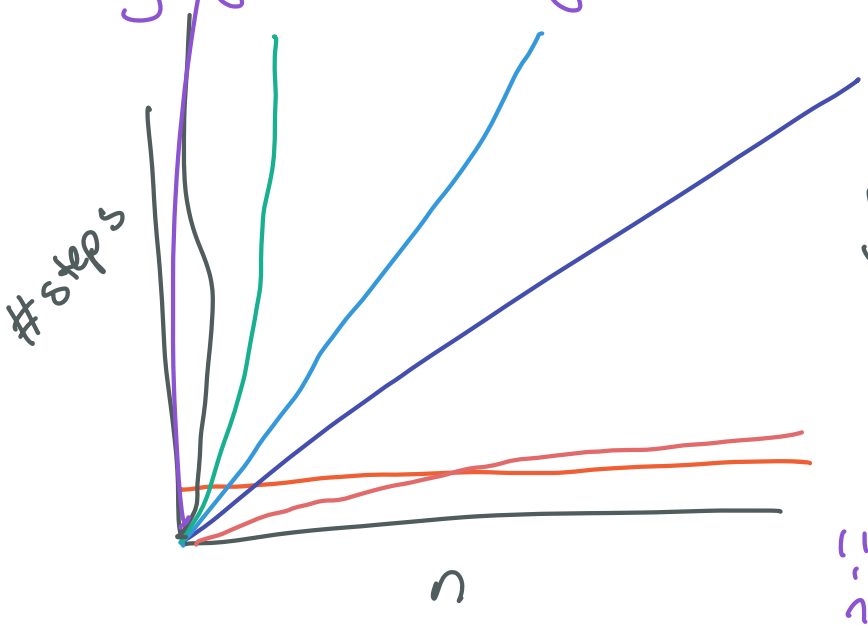
$T'(n) = 3n + 3$

- not identical, but roughly the same run-time



Known, understood complexity classes

↳ any algo should belong to one of them



	$O(1)$	constant
	$O(\lg n)$	logarithmic
!!	$O(n)$	linear
	$O(n \lg n)$	sorting
	$O(n^k)$	polynomial
	.....	
	$O(k^n)$	exponential
!!	$O(n!)$	factorial

↳ intractable

Tractable vs. Intractable

$O(n^2)$

ex slow computer 10,000 ops per second

compare $O(n^2)$ vs. $O(n!)$

$O(n!)$

n	#seconds $- n^2/10,000$
100	1
1,000	100
5,000	2500 (41 mins)
10,000	10,000 (166 mins)
15,000	22,500 (375 mins)

P vs NP ☺

n	#seconds $- n!/10,000$
4	.0024
8	4
10	362
12	47,900 (13 hours)
14	8 mil (100 days)
16	2 bil (66 years)
18	640 bil (202 centuries)