Section 4.1: Big O and Rates of Growth

Tuesday, February 11, 2020 12:07 PM

example: suppose you wish to get digital copies of a number of Star Trek episodes. You can either download them at one episode per hour, or you can get overnight delivery from Amazon (call it 24 hour delivery).

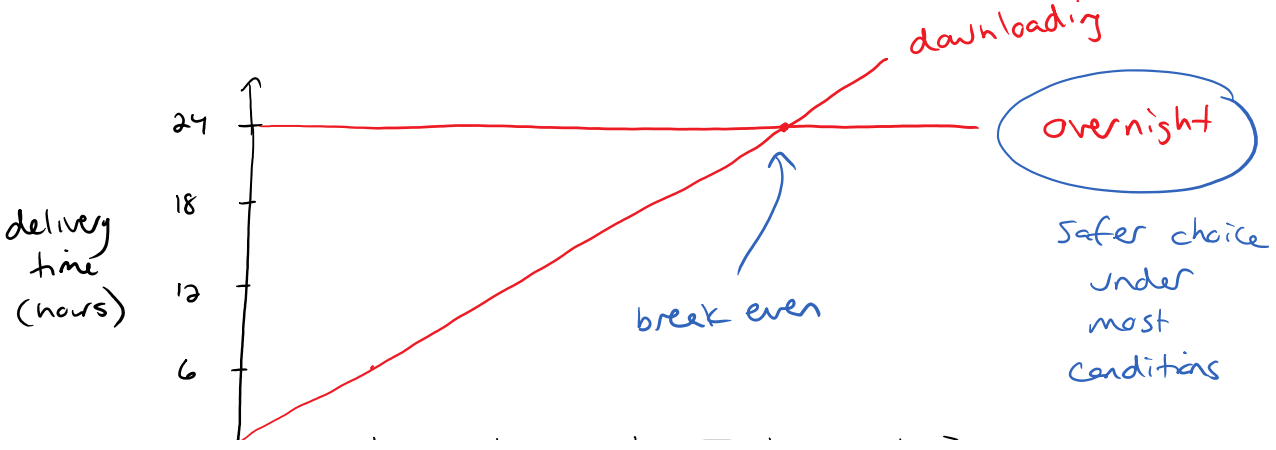
scenarios: we want minimum delivery time (1h/episode vs. 24 hours)

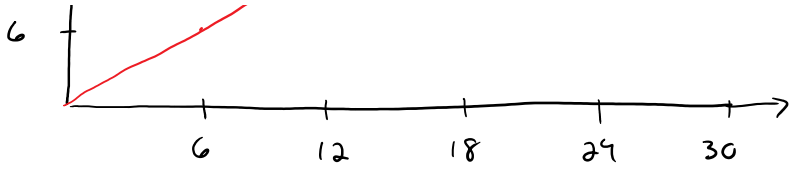
few episodes (< 24), downloading is faster

break even ($= 24$), both methods take same time

many episodes (> 24), overnight is faster

and if you don't know how many episodes then the "many episodes" scenario is the safest approach





2020/02/12

number of steps required for algorithms:

consider adding two 5-digit numbers:

```
  12345
+ 12345
```

} best case: 5 additions

worst case: 5 additions plus some "carries"

Consider multiplying two 5-digit numbers:

```
  12345
× 12345
-------
 xxxxx
 xxxxx
 xxxxx
 xxxxx
 xxxxx
-------
```

} best case:
25 multiplications
plus 9 column additions
and no carries

worst case: many more

the worrying part is the fact that the number of multiplications goes as the <u>square</u> of the

the worrying part is the fact that the number of multiplications goes as the square of the number of digits

100 digits → 10000 multiplications! ouch!

addition: if you double the number of digits, you approximately double the number of operations

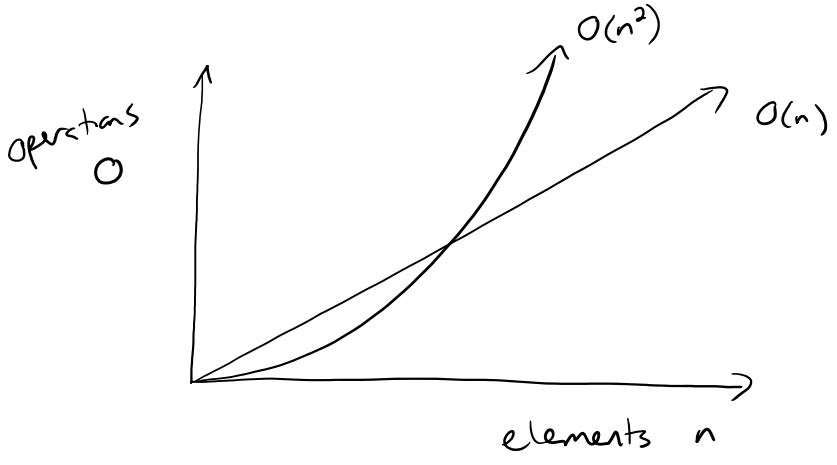
Big O notation: $O(n)$

"order n"

multiplication: if you double the number of digits, you approximately quadruple the number of operations

$O(n^2)$

note: I will not ask you to analyse an algorithm to figure out Big O.



2020/02/13

Big O deals with the number of operations required for large n

(for small n, lines could intersect)

→ for the graph above, for sufficiently large n, $O(n^2)$ is greater than $O(n)$