Complexity and Tractability

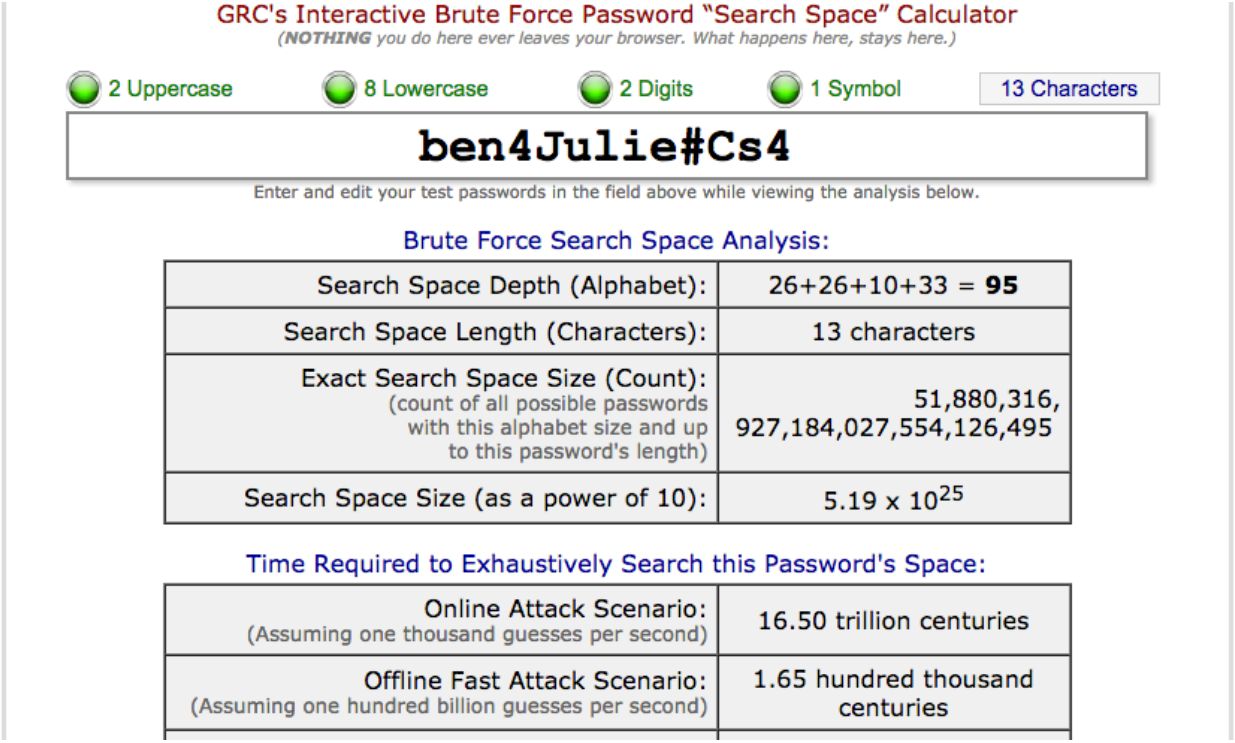
L3 Computer Science

Complexity

- Good to review the L1 information on cost of an algorithm (such as linear vs binary search) to get students thinking about complexity
- Use the field guide to discuss permutations and factorials
- Use the CS field guide widgets and spreadsheet (download) to help them visulise how quickly the cost

www.grc.com/haystack.htm

Use on-line password strength testers for examples of complexity



GRC's Interactive Brute Force Password "Search Space" Calculator

*(**NOTHING** you do here ever leaves your browser. What happens here, stays here.)*

2 Uppercase | 8 Lowercase | 2 Digits | 1 Symbol | 13 Characters

ben4Julie#Cs4

Enter and edit your test passwords in the field above while viewing the analysis below.

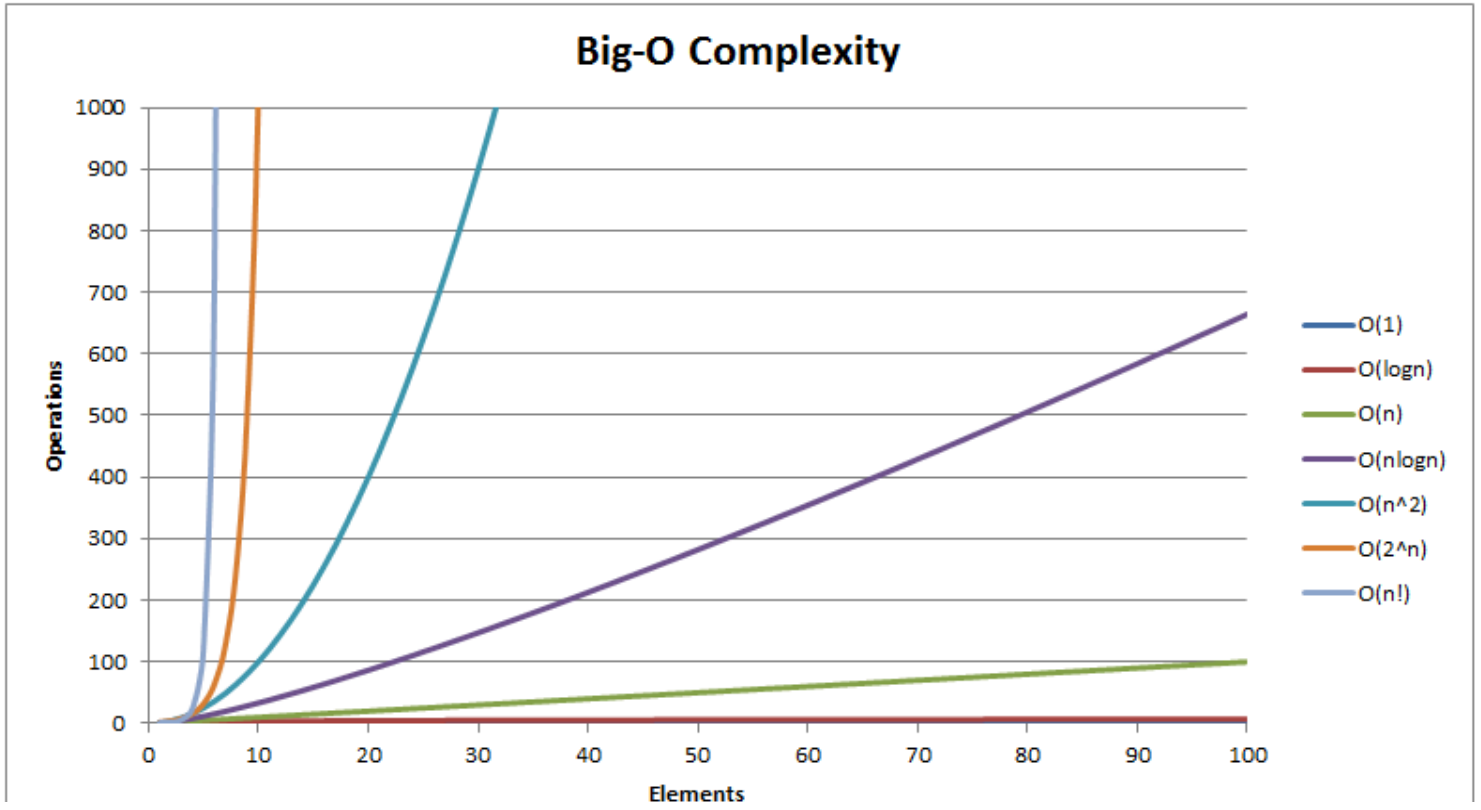
Brute Force Search Space Analysis:

Search Space Depth (Alphabet):	26+26+10+33 = **95**
Search Space Length (Characters):	13 characters
Exact Search Space Size (Count): (count of all possible passwords with this alphabet size and up to this password's length)	51,880,316, 927,184,027,554,126,495
Search Space Size (as a power of 10):	5.19×10^{25}

Time Required to Exhaustively Search this Password's Space:

Online Attack Scenario: (Assuming one thousand guesses per second)	16.50 trillion centuries
Offline Fast Attack Scenario: (Assuming one hundred billion guesses per second)	1.65 hundred thousand centuries

Bia-O Cheat Sheet Chart



Algorithm that take an exponential amount of time or worse for an input of size n, it is labelled as intractable.
Factorial amount of time, n!, is intractable because it's bigger than an exponential function.

Need for Heuristics

- Important to get the students to understand that in the real world a "sub-optimal" but goode enough solution needs to be found to these important problems (TSP, Timetabling, Etc).

Road Trip Examples

http://illuminations.nctm.org/Lesson.aspx?id=2721

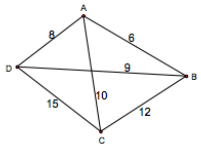
Brute Force

NAME _____

The Brute Force Algorithm

1. Beginning with any starting city, list all the possible round-trips. **Hint:** Draw a tree diagram.
2. Determine the distance of each round-trip.
3. Pick the shortest round-trip.

1. Refer to the map on the right to complete this question.



a) Starting with city A, list all the possible round-trips.

ABCDA
ABDCA
_____
_____
_____
_____

b) Determine the total distance of each round-trip.

ABCDA = AB + BC + CD + DA = 6 + 12 + 15 + 8 = ___
ABDCA = AB + BD + DC + CA = 6 + 9 + __ + __ = ___
ACBDA = _____ = _____
ACDBA = _____ = _____
ADBCA = _____ = _____
ADCBA = _____ = _____

c) Pick the shortest round trip.

Nearest Neighbor

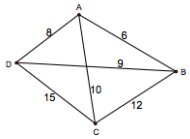
NAME _____

The Nearest Neighbor Algorithm

1. From your starting city, visit the *nearest* city.
2. From that city, visit the *nearest* city you have not already visited.
3. When you have visited all the cities, return to your starting city.

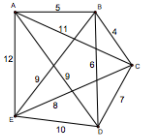
1. Given the table of distances between cities A, B, C, and D and the map, find the shortest round-trip starting at city A.

	A	B	C	D
A	—	6	10	8
B	6	—	12	9
C	10	12	—	15
D	8	9	15	—



A to _____ to _____ to _____ to **A** Total Distance: _____

2. Given the map of cities A, B, C, D, and E, find the length of the round-trip starting at city B using the nearest neighbor algorithm.



B to _____ to _____ to _____ to _____ to **B** Total Distance: _____

Real World Companies

- Telogis (we had a talk from Telogis)
- Kamar - Timetabling is another good example
- Explaining examples of practical applications

BUSINESS TRANSFORMATION
End-to-end solution for company-wide improvements

RAPID DEPLOYMENT
OEM built-in hardware for a fast roll-out

FUEL SAVINGS
Tools to measure, manage & minimize fuel spend

DRIVER SAFETY
Real-time coaching to help drivers self-correct

BETTER PRODUCTIVITY
Uncover the hidden potential of existing resources

AUTOMATED COMPLIANCE
Minimize the compliance burden & reduce downtime

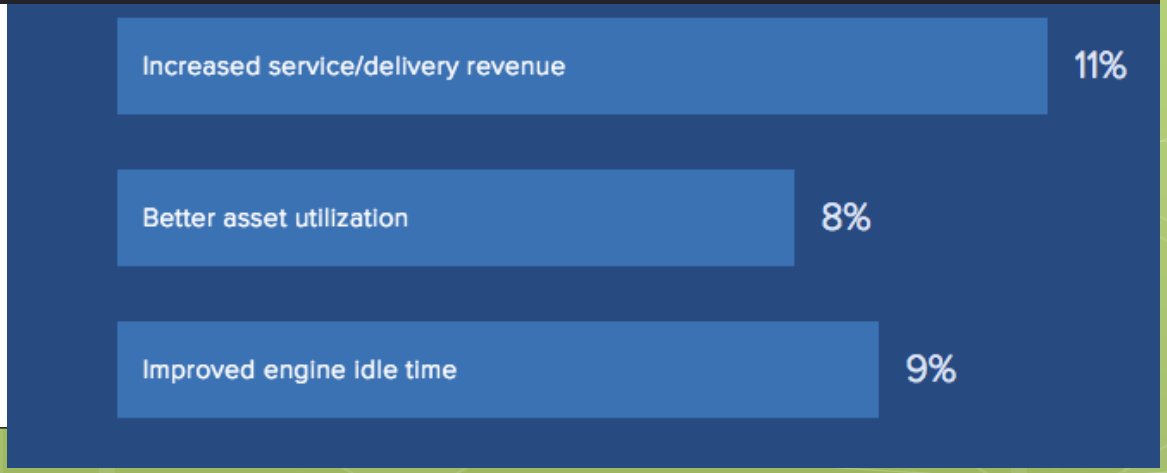
IMPROVED PROFITABILITY
See real savings and better returns across your entire fleet

ASSET SECURITY
Get the best possible return on all assets

SOCIAL RESPONSIBILITY
Boost your brand with safer driving & a greener fleet

FLEXIBLE HARDWARE
Choose the hardware that fits your business best

REDUCED INSURANCE
Improved safety & fewer claims reduces premiums



Compare EcoMaps to RouteXL

- Through testing the two apps side by side students (and reading the FAQ/About on Route XL) can draw conclusions about whether a brute force (intractable) or heursitic is being used.
- Students can evaluate the effectiveness of the two apps (Exc)
- Discuss examples of practical applications (Exc)

Encryption

- Encryption and HUGE numbers – Numberphile (good video)
- www.youtube.com/watch?v=M7kEpw1tn50
- Students can use this as another practical application to discuss and to evaluate the effectiveness
- Can summarise with a good evaluation of what the positive and negatives and implications of intractable algorithms (e.g. TSP problems vs encryption)

Points of student confusion

- Interchanging complicated and complex (complexity is a specific word related to the time complexity of solving the problem)
- Tractable problems can't become intractable (e.g. using 5 data points for the TSP doesn't mean the problem is tractable - intractable problems are only solvable for very small data sets).
- Intractable algorithms do not get more complex: (n-1)!/2 doesn't change.