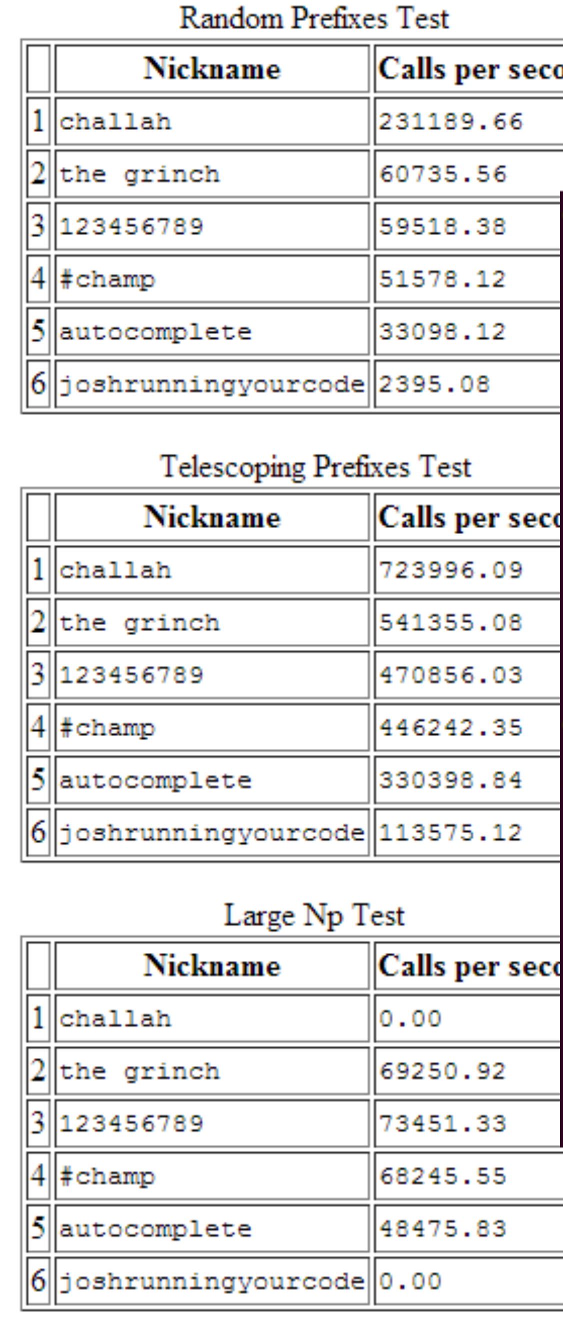
Better living through functional programming (COS 326)



Random Prefixes Test

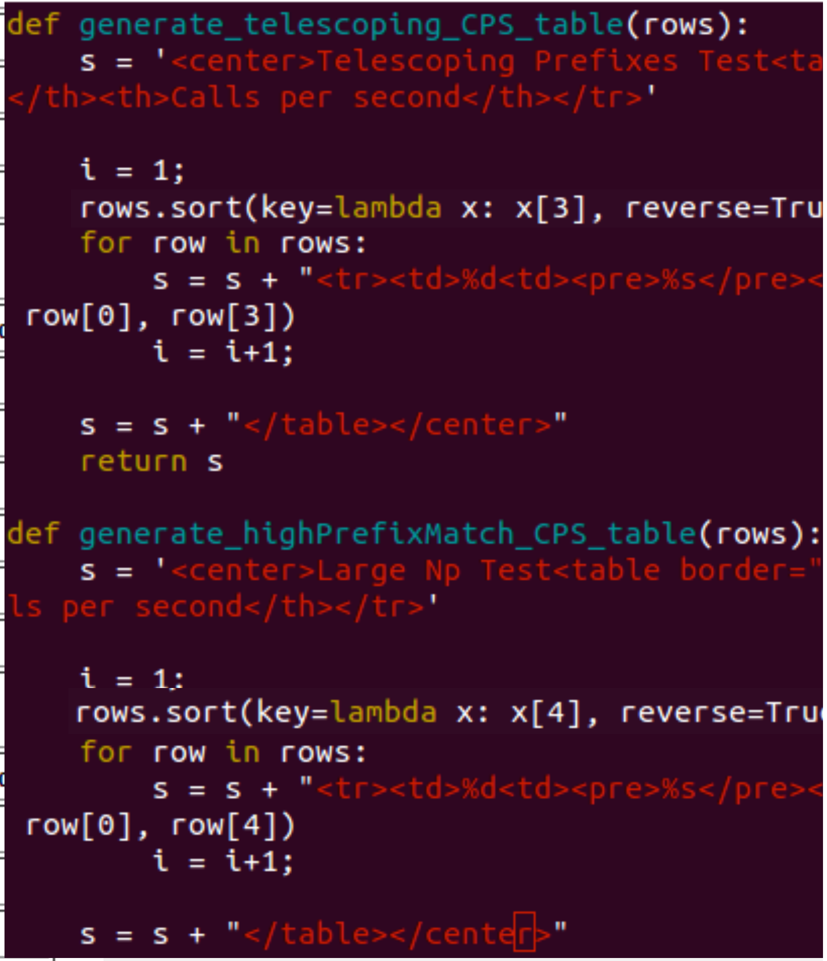
	Nickname	Calls per second
1	challah	231189.66
2	the grinch	60735.56
3	123456789	59518.38
4	#champ	51578.12
5	autocomplete	33098.12
6	joshrunningyourcode	2395.08

Telescoping Prefixes Test

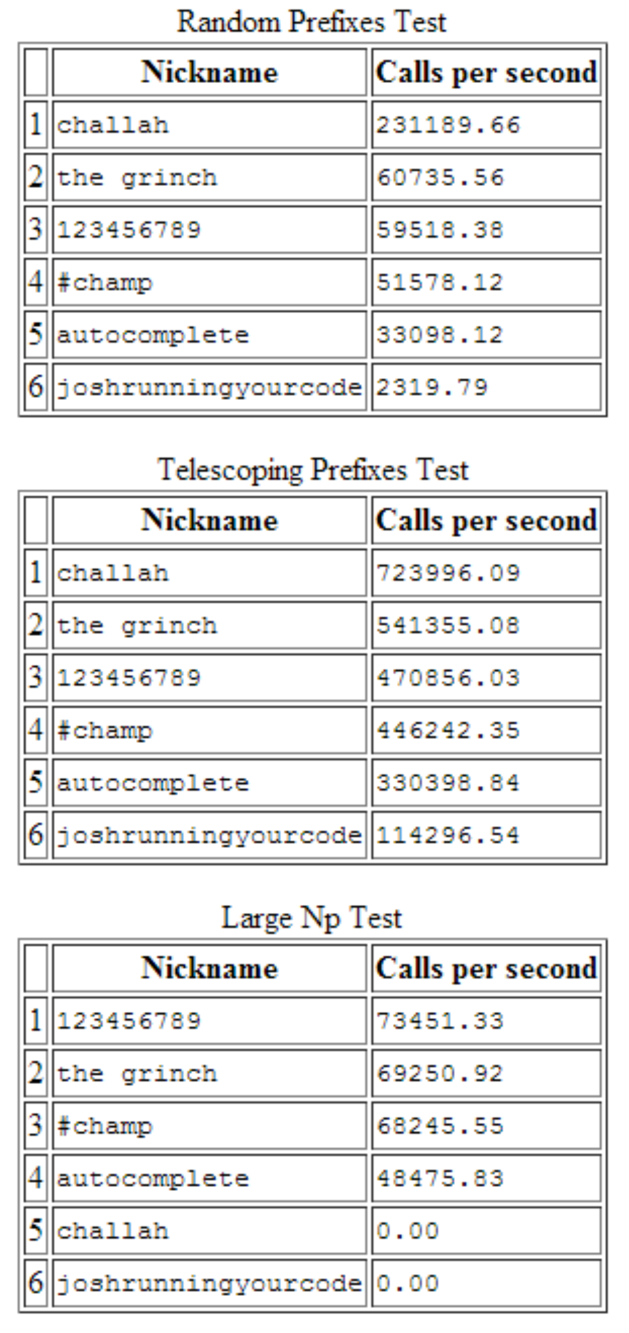
	Nickname	Calls per seco
1	challah	723996.09
2	the grinch	541355.08
3	123456789	470856.03
4	#champ	446242.35
5	autocomplete	330398.84
6	joshrunningyourcode	113575.12

Large Np Test

	Nickname	Calls per seco
1	challah	0.00
2	the grinch	69250.92
3	123456789	73451.33
4	#champ	68245.55
5	autocomplete	48475.83
6	joshrunningyourcode	0.00



```
def generate_telescoping_CPS_table(rows):
    s = '<center>Telescoping Prefixes Test<ta
</th><th>Calls per second</th></tr>'

    i = 1;
    rows.sort(key=lambda x: x[3], reverse=Tru
    for row in rows:
        s = s + "<tr><td>%d<td><pre>%s</pre><
 row[0], row[3])
        i = i+1;

    s = s + "</table></center>"
    return s

def generate_highPrefixMatch_CPS_table(rows):
    s = '<center>Large Np Test<table border="
ls per second</th></tr>'

    i = 1;
    rows.sort(key=lambda x: x[4], reverse=Tru
    for row in rows:
        s = s + "<tr><td>%d<td><pre>%s</pre><
 row[0], row[4])
        i = i+1;

    s = s + "</table></center>"
```



Random Prefixes Test

	Nickname	Calls per second
1	challah	231189.66
2	the grinch	60735.56
3	123456789	59518.38
4	#champ	51578.12
5	autocomplete	33098.12
6	joshrunningyourcode	2319.79

Telescoping Prefixes Test

	Nickname	Calls per second
1	challah	723996.09
2	the grinch	541355.08
3	123456789	470856.03
4	#champ	446242.35
5	autocomplete	330398.84
6	joshrunningyourcode	114296.54

Large Np Test

	Nickname	Calls per second
1	123456789	73451.33
2	the grinch	69250.92
3	#champ	68245.55
4	autocomplete	48475.83
5	challah	0.00
6	joshrunningyourcode	0.00

Algorithms

Robert Sedgewick | Kevin Wayne

http://algs4.cs.princeton.edu

Really Hard Problems

- *Dealing with NP (or harder) problems*
- *What if P=NP?*
- *Beyond complexity classes*

Birds-eye view: review

Desiderata. Classify problems according to computational requirements.

complexity	order of growth	examples
linear	N	min, max, median, Burrows-Wheeler transform, ...
linearithmic	N log N	sorting, element distinctness, convex hull, closest pair, ...
quadratic	N^2	?
⋮	⋮	⋮
exponential	c^N	?

Frustrating news. Huge number of problems have defied classification.

Birds-eye view: revised

Desiderata. Classify problems according to computational requirements.

complexity	order of growth	examples
linear	N	min, max, median,
linearithmic	N log N	sorting, convex hull,
M(N)	?	integer multiplication, division, square root, ...
MM(N)	?	matrix multiplication, Ax = b, least square, determinant, ...
⋮	⋮	⋮
NP-complete	probably not N^b	3-SAT, IND-SET, ILP, ...

Good news. Can put many problems into equivalence classes.

Handling NP Completeness

The bad news

- You’ve proven that your problem of interest is NP Complete.

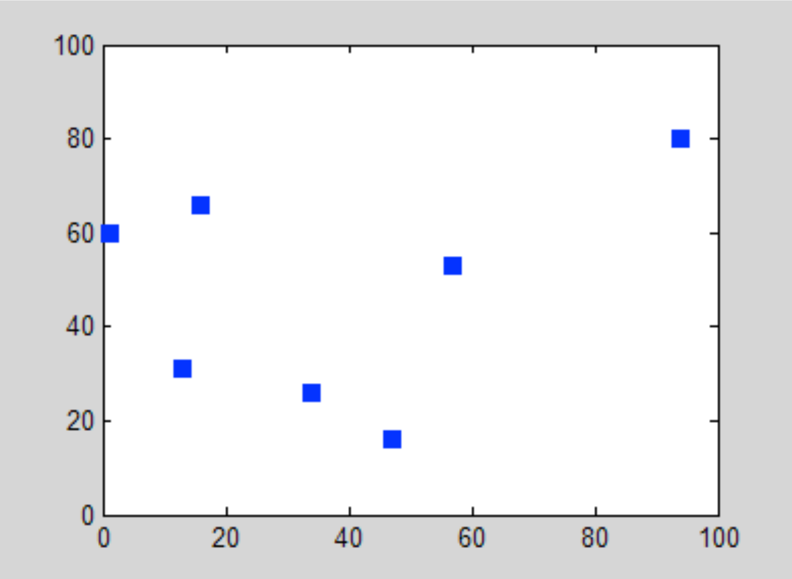
The good news

- There are a few things you can try:
  - Accept a non-optimal answer.
  - Prune the search space.
  - Take advantage of special structure in the problem.

Approximation

Approach 1: Approximate Heuristics

- Accept incorrect answers
  - TSP, always go to closest city next

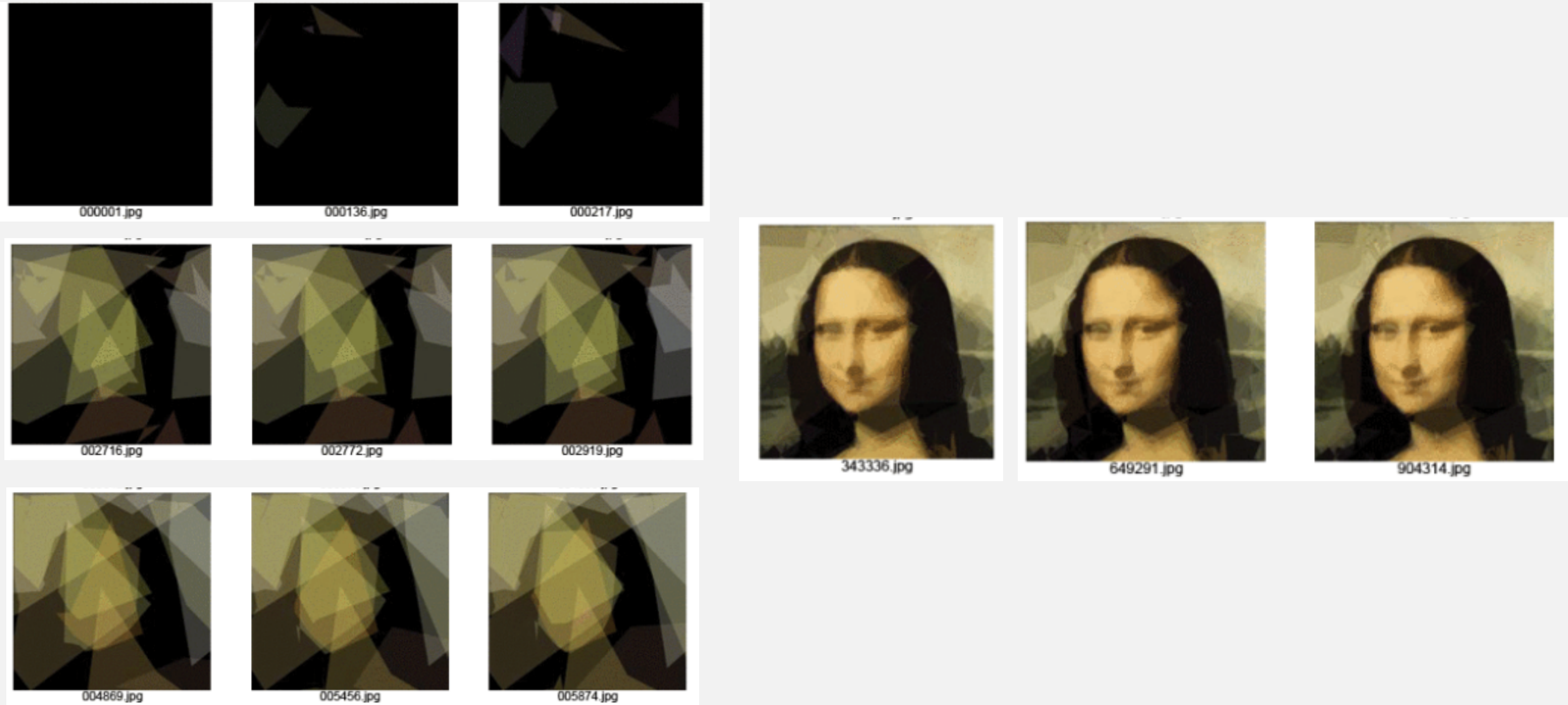


http://en.wikipedia.org/wiki/Travelling_salesman_problem

Approximation

Approach 1: Approximate Heuristics

- Accept incorrect answers
  - Polygon approximation problem: Find colored 50 triangles that best represent an image. Use genetic algorithm to make choice.

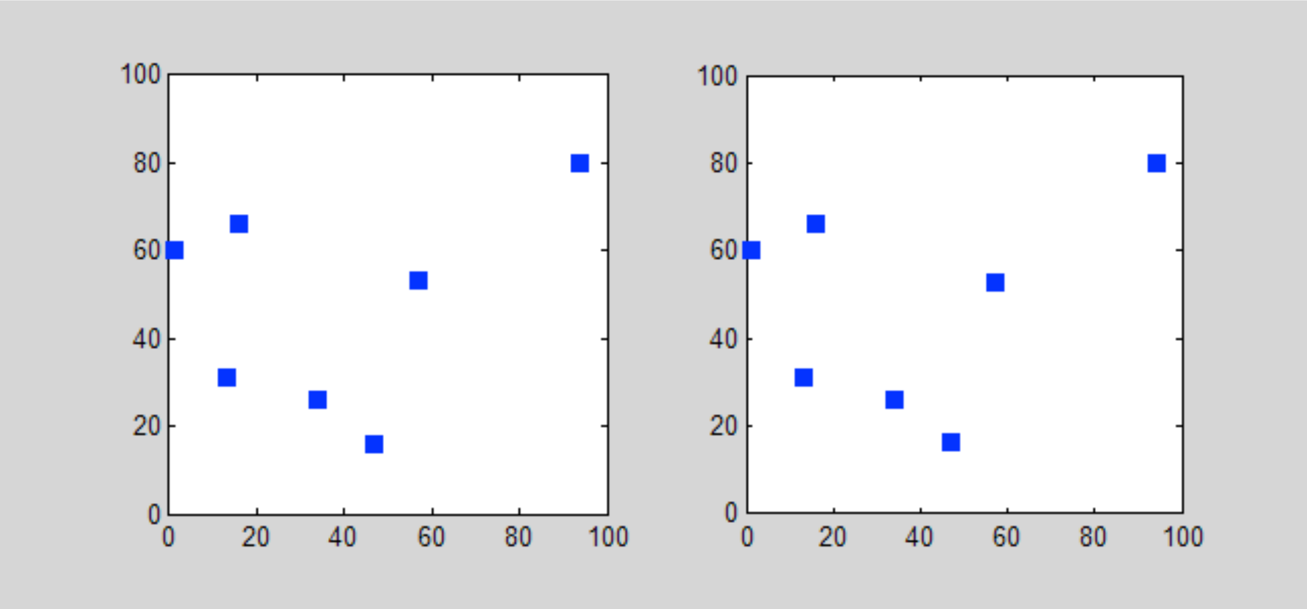


http://rogeralsing.com/2008/12/07/genetic-programming-evolution-of-mona-lisa/

Smarter Searching

Approach 2: Exact Heuristics

- Use a smarter (but still worst case intractable) solution technique
  - TSP: Brute force, but rule out clearly inferior solutions to best known solutions (branch and bound)



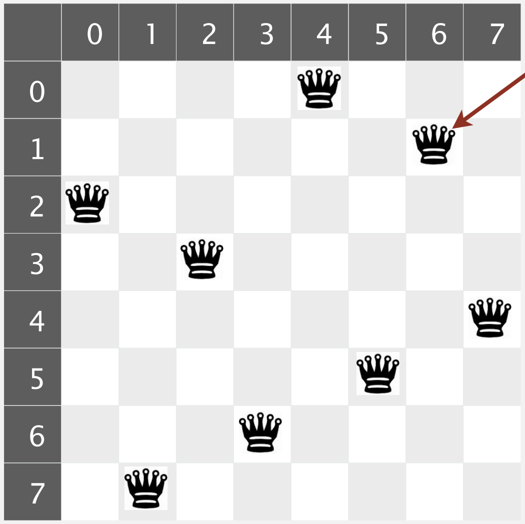
Checks 129 out of 360 possible permutations

http://en.wikipedia.org/wiki/Travelling_salesman_problem

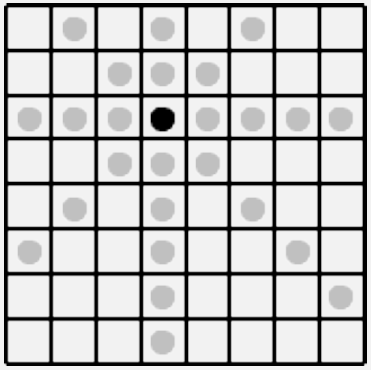
Decision problem is not in NP

Smarter Searching: N-queens problem

Q. How many ways are there to place N queens on an N-by-N board so that no queen can attack any other?



a[1] = 6 means the queen from row 1 is in column 6



```
int[] a = { 2, 7, 3, 6, 0, 5, 1, 4 };
```

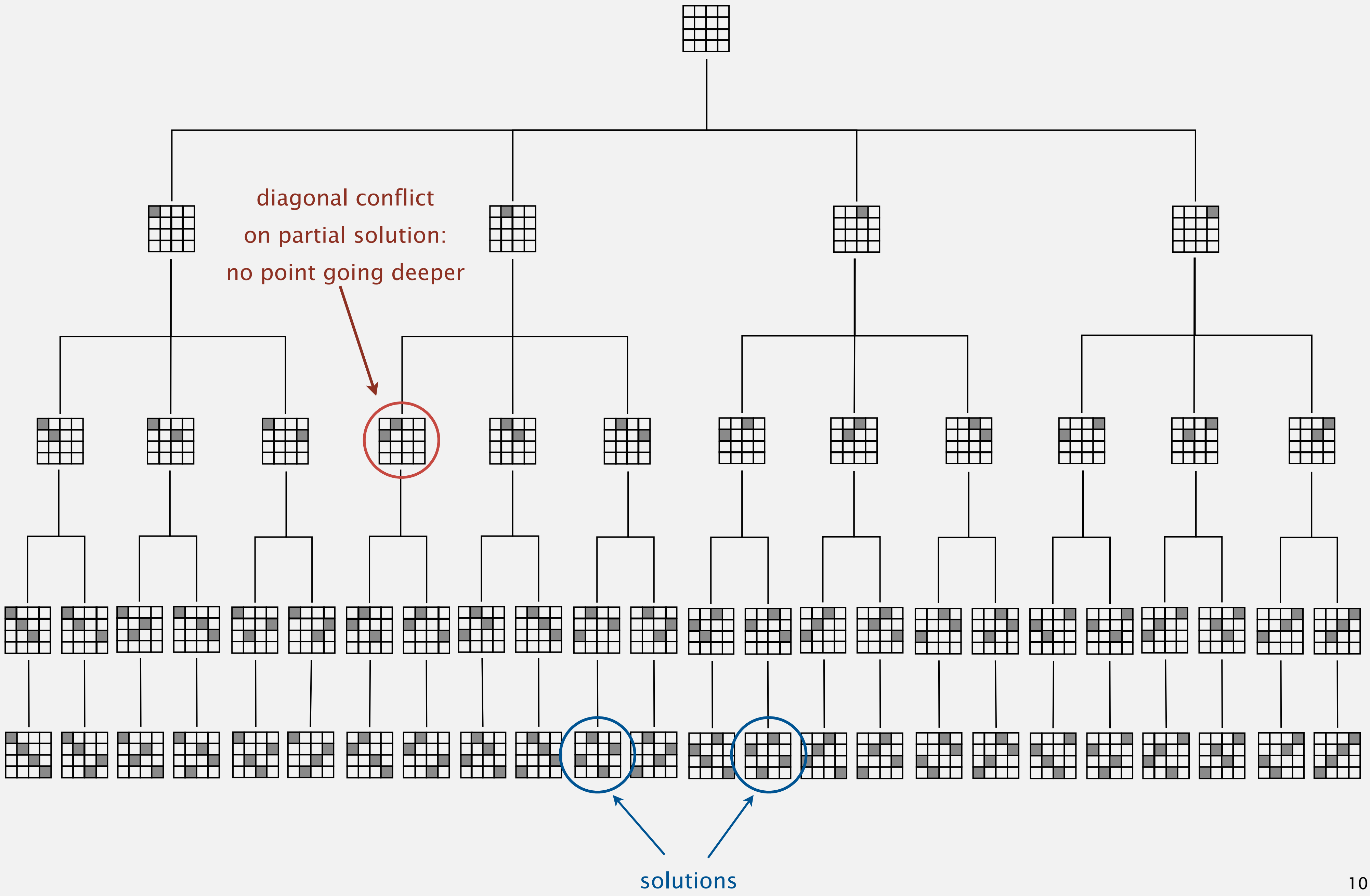
Representation. No 2 queens in the same row or column ⇒ permutation.

Additional constraint. No diagonal attack is possible.

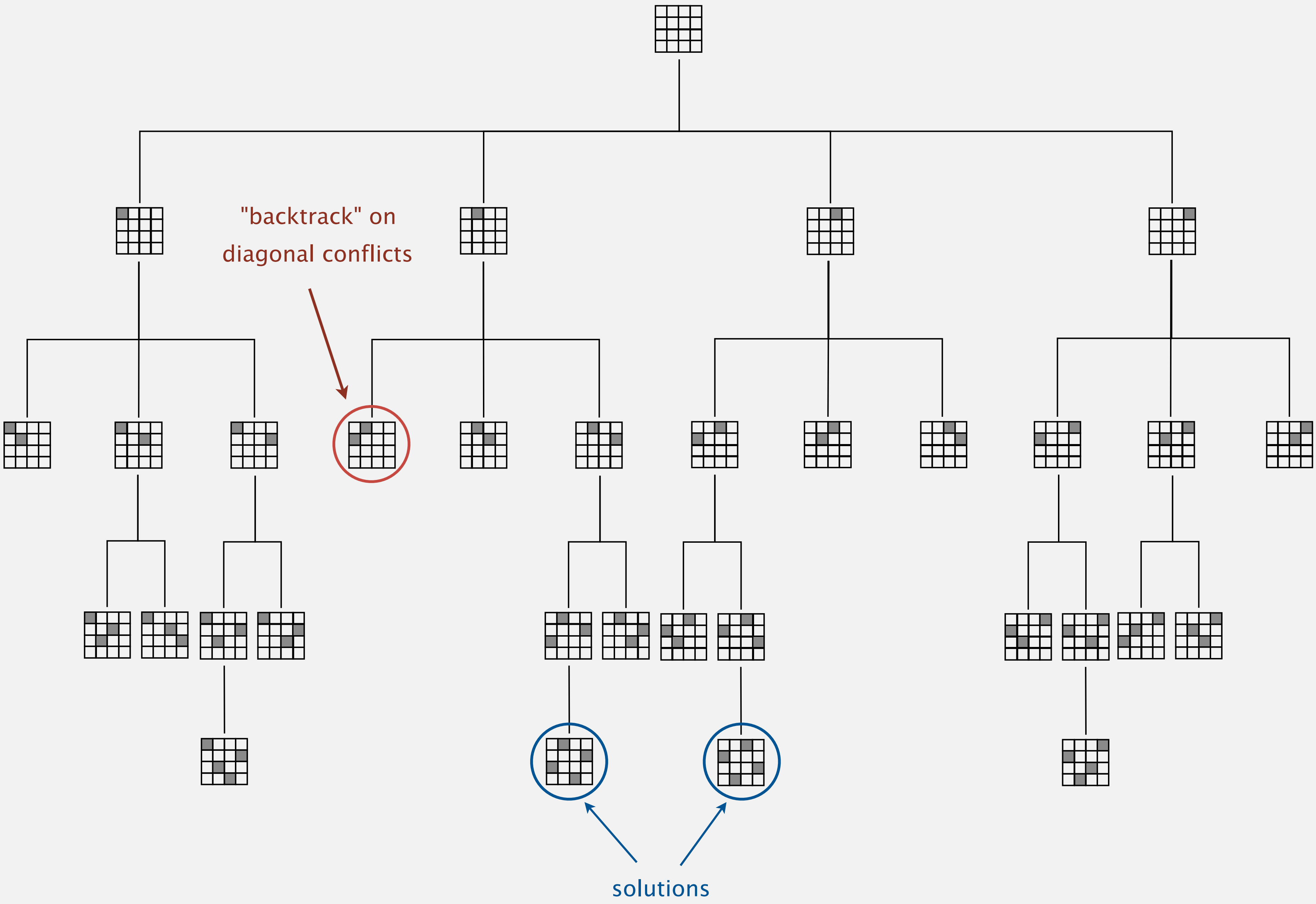
Challenge. Enumerate (or even count) the solutions. ←

unlike N-rooks problem, nobody knows answer for N > 30

4-queens search tree



4-queens search tree (pruned)



Wake up

Approach 3:

- Wake up and realize that it was all just a dream.

Dealing with intractable problems

Approach 4: Take advantage of special structure

- Realize that your problem is actually a special, solvable case.

Example

- Hamiltonian Path
  - NP-Complete in general.
  - Easy to show that it's in P if the graph is a straight line or tree.

Algorithms

ROBERT SEDGEWICK | KEVIN WAYNE

http://algs4.cs.princeton.edu

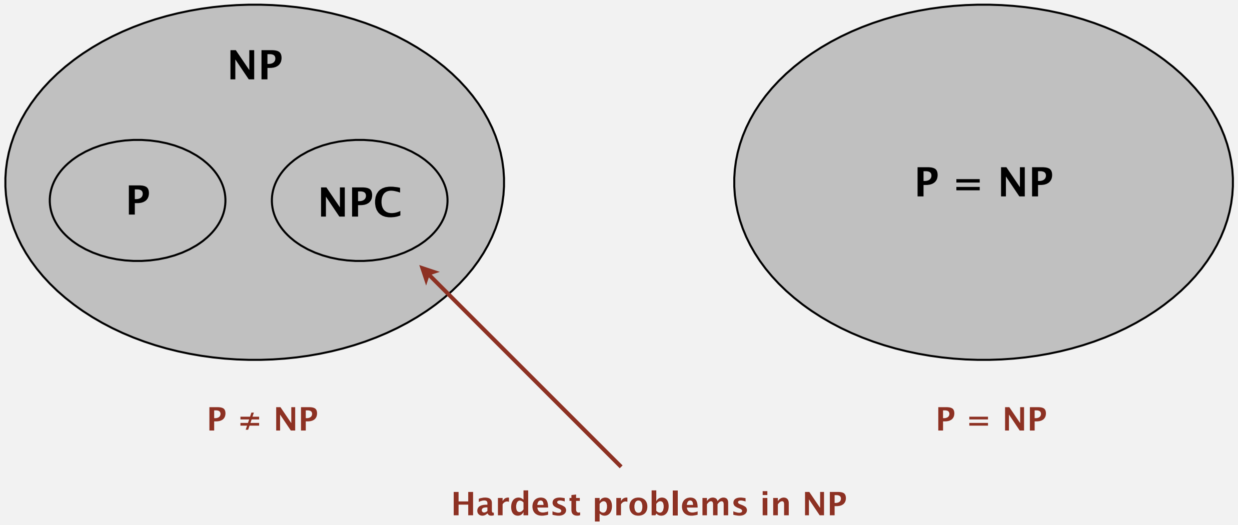
REALLY HARD PROBLEMS

- *Dealing with NP (or harder) problems*
- *What if P=NP?*
- *Beyond complexity classes*

P = NP?

Does P = NP?

- Equivalently: Is any NP Complete problem also in P?
- Equivalently: Efficiently verifiable ⇒ efficiently solvable?

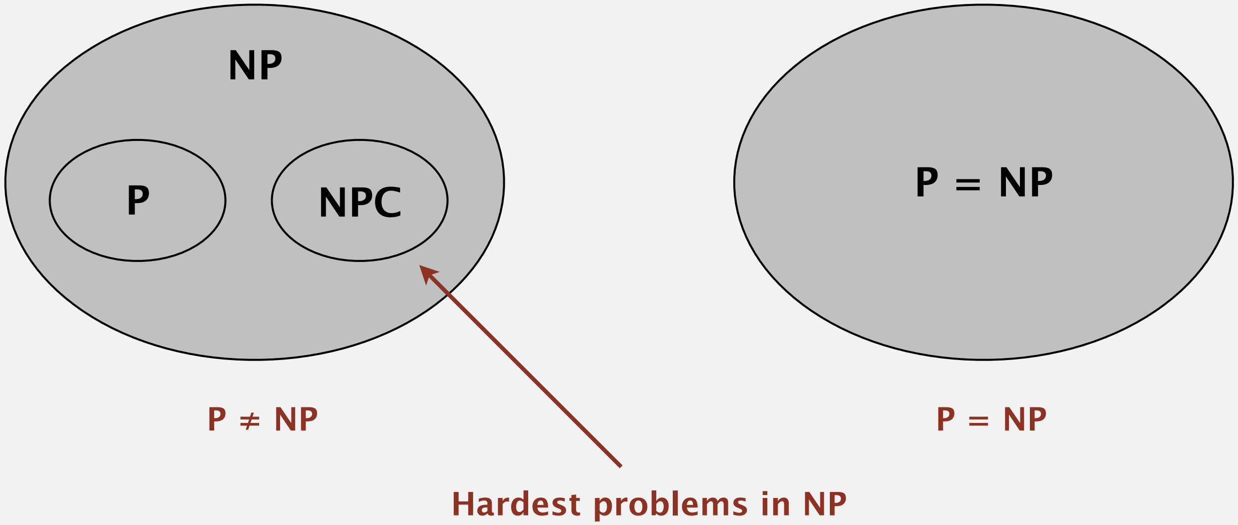


Reminder: NP may as well have been called VP for “Verifiable in Polynomial Time”

P = NP?

Does P = NP?

- Equivalently: Is any NP Complete problem also in P?
- Equivalently: Efficiently verifiable ⇒ efficiently solvable?



P = NP?

Consensus Opinion (Bill Gasarch poll, 2002)

- 9 - Yes
- 61 - No
- 4 - Independent of axiomatic systems typically used in considering the problem.

Why is opinion generally negative?

- Someone would have proved it by now.
  - "The only supporting arguments I can offer are the failure of all efforts to place specific NP-complete problems in P by constructing polynomial-time algorithms." - Dick Karp
  - "For God's sake, let's keep it open for another 100 years! NSF needs to be convinced that theoretical CS is still relevant and supports it." - Ming Li
- Creation of solutions seems philosophically more difficult than verification.
- Mathematical reasons: Way beyond scope of course (and my understanding)

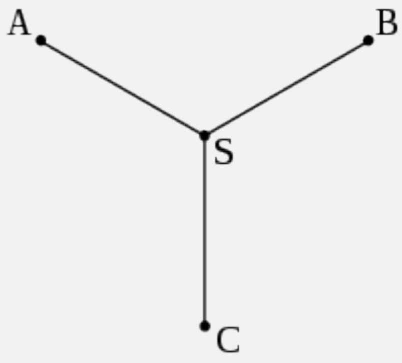
Solving NP Complete Problems

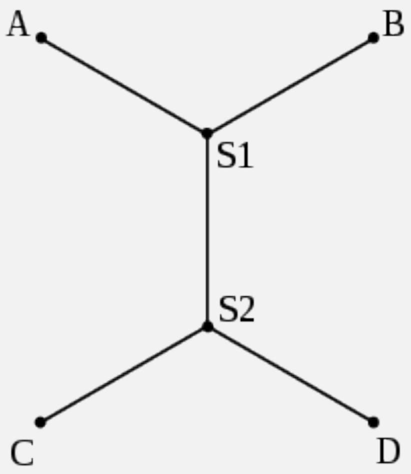
One approach

- Does nature solve any NP complete problems?
- If so, do any such problems admit a polynomial time simulator?

Classic Example

- Soap bubbles and Steiner Trees.





Like Minimum Spanning Trees but you can make up new nodes!

Soap Bubbles and Steiner Trees

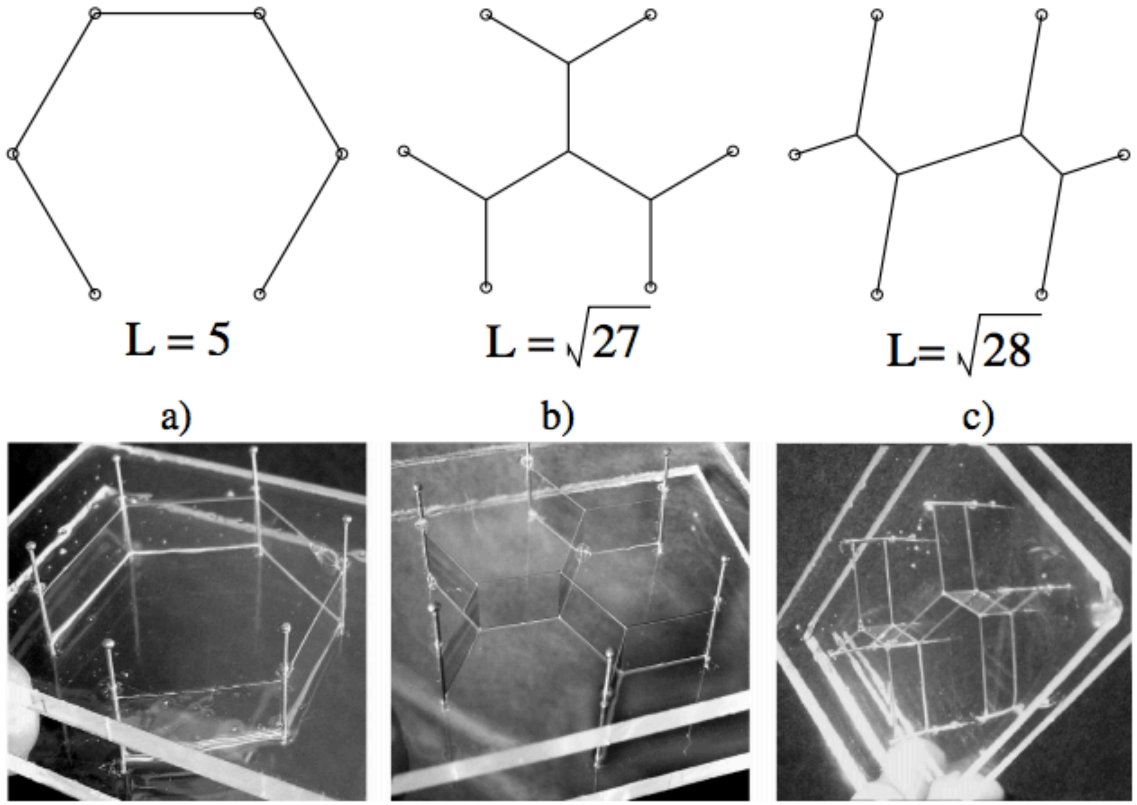


Figure 2: Commonly reported soap film configurations for 6-pin regular hexagon.

http://arxiv.org/pdf/0806.1340v1.pdf

Soap Bubbles mean P=NP?

> *The Clay Mathematics Institute offers a $1 million prize for a solution to the P=?NP problem. We look forward to receiving our award — but concede that the expected format of a solution is an object-level proof, not a meta-level argument like what we provide...*
>
> *... the fact of the matter is that the analog process we exploit is a painfully simple macroscopic phenomenon — as we say, a "normal" physical process. The burden of proof is surely on those who would maintain that the formal machinery of digital physics is insufficient to model something as straightforward as submerging nails in, and retrieving them from, a bucket of soapy water.." — Bringsjord and Taylor (2004)*

http://arxiv.org/pdf/cs/0406056.pdf

The problem

- Soap bubbles aren't always right.
  - Fall into local optima just like heuristic approaches to finding Steiner trees.

Protein Folding

The HP model of protein folding is NP-complete.

- Incorrect but useful analogy: Imagine a shoelace with a bunch of positive and negative charges stuck to various points.
- Simulation tries to figure out final state of charged shoelace if you shake it and let it float into space.

*

Mistake in simulation

- Wrong shape.
- Wrong function prediction.

Mistakes occur in nature

- Bad news for cows (mad cow disease!)
- Bad news for computing (Billions of years of evolution. No solution!)
- Bad news for you.

*: http://themisadventuresofamisplacedalaskan.blogspot.com/2012/04/mad-cow-strikes-back.html

But what if P = NP?

> *"[A linear or quadratic-time procedure for what we now call NP-complete problems would have] consequences of the greatest magnitude. [For such an procedure] would clearly indicate that, despite the unsolvability of the Entscheidungsproblem, the mental effort of the mathematician in the case of yes-or-no questions could be completely replaced by machines." — Kurt Gödel*

One of these things, is not like the other..

Millenium Prize Problems

- Hodge Conjecture
- Poincare Conjecture (solved!)
- Riemann Hypothesis
- Yang-Mills existence and mass gap
- Navier-Stokes existence and smoothness
- Birch and Swinnerton-dyer conjecture
- P=NP
  - If true, proof might allow you to trivially solve all of the other problems.

What if P = NP?

"I have heard it said, with a straight face, that a proof of P = NP would be important because it would let airlines schedule their flights better, or shipping companies pack more boxes in their trucks!

If [P = NP], then we could quickly find the smallest Boolean circuits that output (say) a table of historical stock market data, or the human genome, or the complete works of Shakespeare. It seems entirely conceivable that, by analyzing these circuits, we could make an easy fortune on Wall Street, or retrace evolution, or even generate Shakespeare's 38th play. For broadly speaking, that which we can compress we can understand, and that which we can understand we can predict.

So if we could solve the general case—if knowing something was tantamount to knowing the shortest efficient description of it—then we would be almost like gods. [Assuming P ≠ NP] is the belief that such power will be forever beyond our reach." — Scott Aaronson

http://www.scottaaronson.com/papers/npcomplete.pdf

Great paper: Also where I stole the ideas about soap bubbles and protein folding.

For more: **The Golden Ticket: P, NP, and the Search for the Impossible by Lance Fortnow**

Algorithms

Robert Sedgewick | Kevin Wayne

http://algs4.cs.princeton.edu

Really Hard Problems

- *Dealing with NP (or harder) problems*
- *What if P=NP?*
- *Beyond complexity classes*

Can machines think?

AI-Complete problems

- Communication in natural language
- Translation of natural language
- Writing a play
- Programming
- Theorem proving (can be formalized into an NP complete problem)

Can machines think?

AI-Complete problems

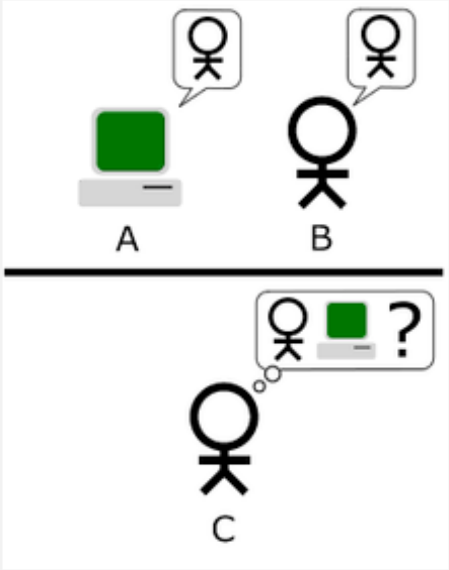
- Communication in natural language
- Translation of natural language
- Writing a play
- Programming
- Theorem proving (can be formalized into an NP complete problem)

Proof of concept

- Brains
- Bad API

The Turing test

Separate the scientific question from the philosophical question



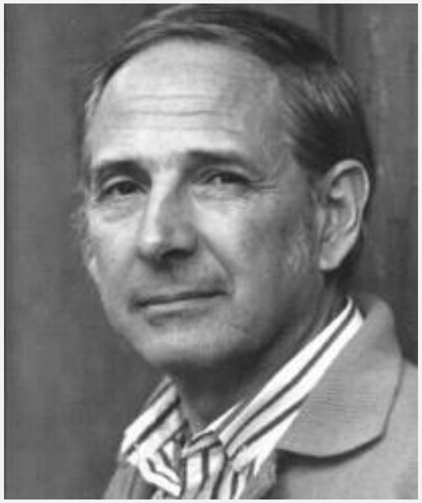
http://cmst1a0970733.blogspot.com/2009/10/collective-computer-intelligence-mark.html

http://www.cosc.canterbury.ac.nz/csfieldguide/student/Artificial%20intelligence.html

Searle's Chinese Room

John Searle

- Incredibly famous professor in the philosophy of mind (and language).
- Goal:
  - Debunk the Turing test (and similar) as proof of intelligence.



Searle's Chinese Room

Imagine a box with a computer inside.

- You slip messages in Chinese into a slot in a box.
- The machine reads the Chinese, processes it according to some set of rules, and prints out a response.
- The Chinese room behaves exactly as if it were a native Chinese speaker.
- Does the machine truly understand Chinese (Strong AI), or is it just simulating understanding (weak AI)?

我爸是李刚!

Karl Gottlieb von Windisch's 1784 book *Inanimate Reason*

Searle’s Chinese Room

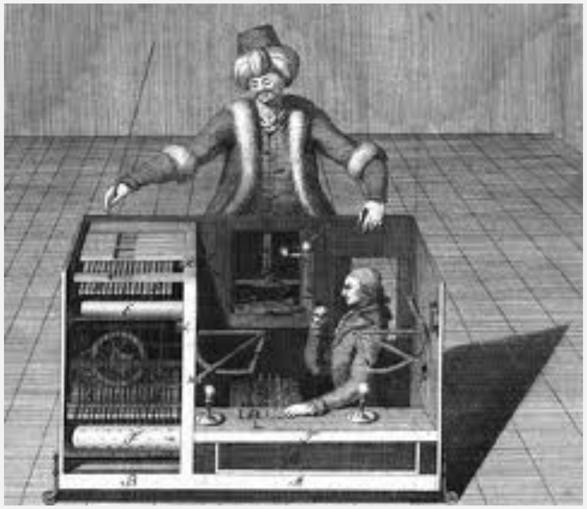
Observation

- The human mind can simulate a Turing machine (equivalent to: it is possible to manually simulate Java code).

Searle's Chinese Room

Imagine a box with a guy inside.

- You slip messages in Chinese to a monolingual Anglophone in a box.
- The guy has:
  - A printout of the source code of the Chinese room computer program.
  - An enormous supply of pencils, erasers, paper, and filing cabinets.
- When he receives a message, the guy mechanically follows the rules, writes out the resulting response, and slips it back to you.
- Since the guy in the box doesn't understand Chinese (or what he's saying), then the computer doesn't either: it's just simulating intelligence.

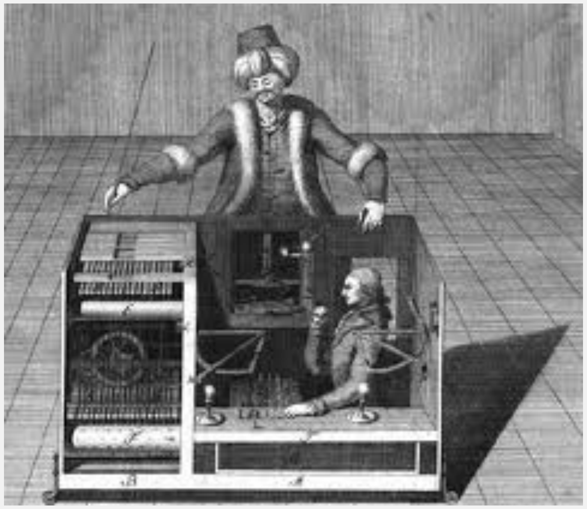


Karl Gottlieb von Windisch's 1784 book *Inanimate Reason*

Josh's Searle's Chinese Room

Imagine a box with a humanoid robot inside.

- You slip messages in Chinese to a humanoid robot inside a box.
- The robot has:
  - A printout of the source code of the Chinese room computer program.
  - An enormous supply of pencils, erasers, paper, and filing cabinets.
  - Minimal complexity: Ability to OCR characters and follow rules in book.
- When he receives a message, the robot mechanically follows the rules, writes out the resulting response, and slips it back to you.
- Does the humanoid robot understand Chinese?
  - My opinion: No, and it's irrelevant!



The Chinese Room

What the Chinese Room Purports to Prove

- The Turing Test doesn't prove intelligence.
- Argument:
  - Turing: If a digital computer mechanically generates good conversation, then the computer IS sentient.
  - Searle: Since a man can unthinkingly generate good conversation according to same mechanical rules, then a computer doing the same thing is NOT sentient (because the man wasn't thinking).

- My opinion: The man in the Chinese Room is acting as a non-sentient component of a sentient system.
  - The 'smarts' of the system are all in the rulebook (which is going to be fantastically complex).

> *"[The field of cognitive science ought to be redefined as] the ongoing research program of showing Searle's Chinese Room Argument to be false."* — *Pat Hayes*

Can humans think?

Possible consequences if the Chinese Room (or similar) arguments are right

- Just because YOUR brain is providing me with a good conversation doesn't prove anything!
- You might just be pretending to be sentient.
  - Philosophical zombie.

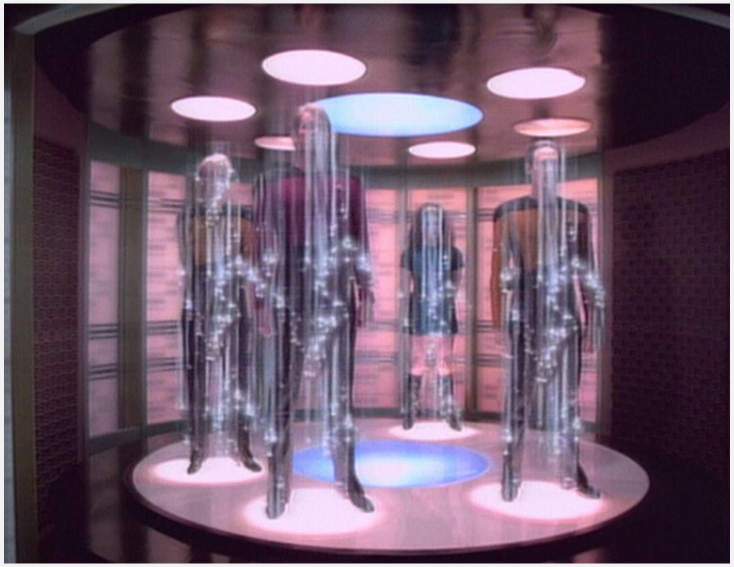
> *“**Intellect:** By convention there is sweetness, by convention bitterness, by convention color, in reality only atoms and the void.*
>
> ***Senses:** Foolish intellect! Do you seek to overthrow us, while it is from us that you take your evidence?” — Democritus (circa 450 BC), dialogue between the intellect of senses*

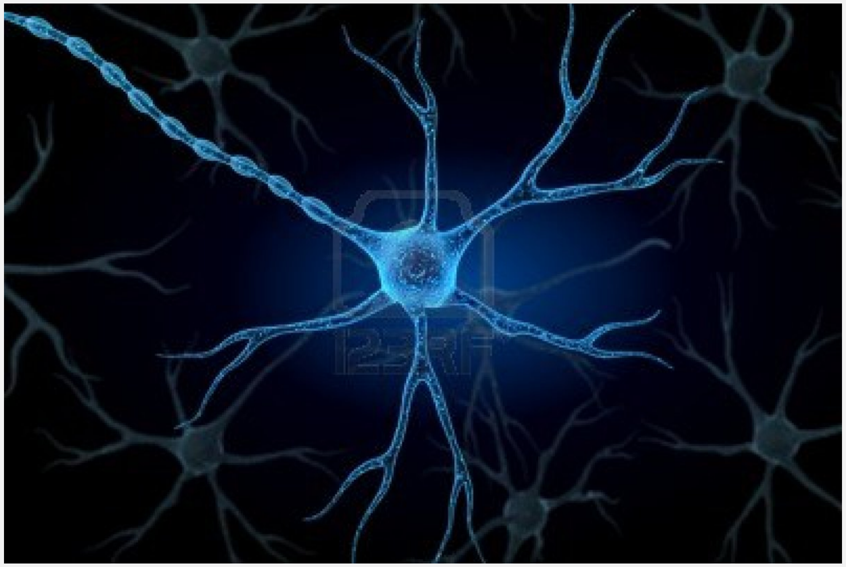
Quote by way of Scott Aaronson (in turn by way of Schrodinger)

The Hug Paradox

Josh Hug believes brains are just mechanical machines.

- Josh Hug would also not volunteer to be deconstructed and then reconstructed.
  - He would be perfectly willing to have his brains cells replaced by precise electronic analogues, one by one, at a reasonable rate (say entire brain in one day).
  - If replacement process was done in big enough chunks: Hell no!
  - There is clearly an error in Josh Hug's thinking.





http://us.123rf.com/400wm/400/400/rbhavana/rbhavana1012/rbhavana101200549/8368432-neuron-in-isolated-background.jpg

Deep into the fringes

The Chinese Lookup Table

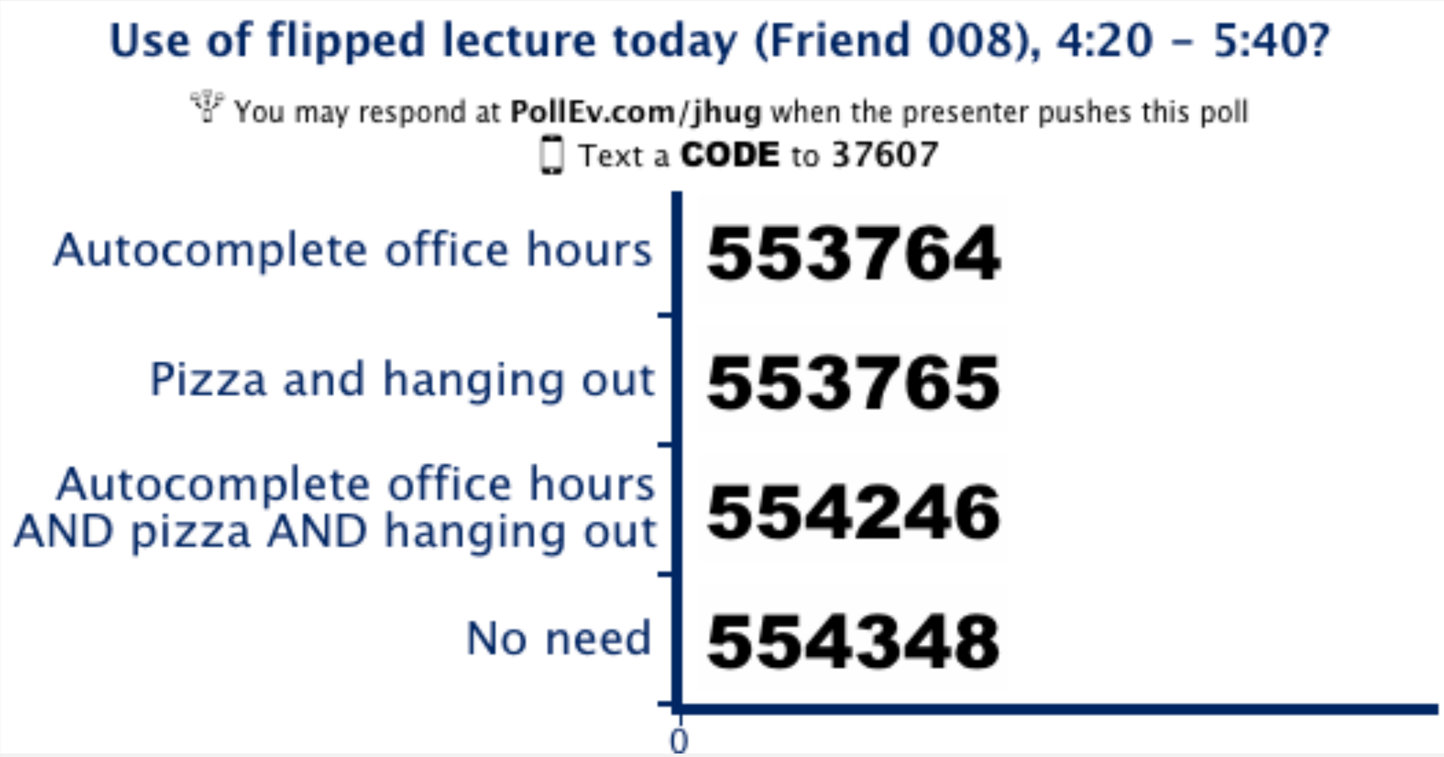
- Suppose we have an enormous lookup table that encodes every possible stream of Chinese conversation of duration 20 years or less.
- Features:
  - More atoms than the universe (obscenely so!).
  - Indistinguishable from a sentient being (for 20 years at least).
- Would such a lookup table think?

Things to ponder

- Lookup in a symbol table is boring (i.e. simple operation).
  - Gut feeling: Doesn't seem like this could imply consciousness.
- Kolmogorov complexity of this lookup table is massive compared to a representation of human intelligence.
  - Consciousness and compression are somehow linked [??].
- Chance of generating a good lookup table randomly is non-zero!

Disclaimer: This is probably a common concept in the Philosophy of Mind.
I didn't do my homework to see what people think about this.

Last poll everywhere question



Use of flipped lecture today (Friend 008), 4:20 – 5:40?

You may respond at **PollEv.com/jhug** when the presenter pushes this poll

Text a **CODE** to 37607

Autocomplete office hours	**553764**
Pizza and hanging out	**553765**
Autocomplete office hours AND pizza AND hanging out	**554246**
No need	**554348**