Class 16: NP-Completeness / The Story so Far

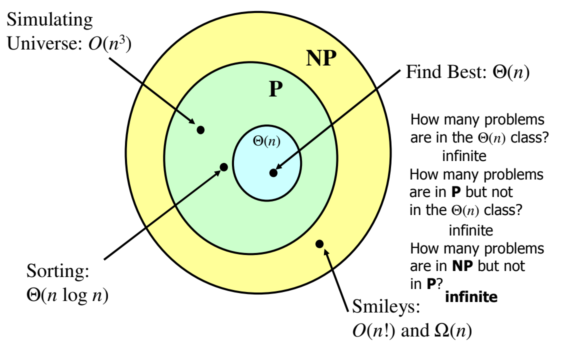
CS150: Computer Science
University of Virginia
Computer Science

David Evans
http://www.cs.virginia.edu/evans

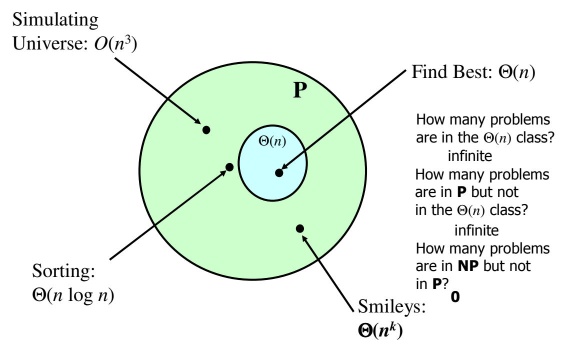
Menu

- 3SAT
- Complexity class NP-Complete
- The Story so Far
- Some NP-Complete Problems

Problem Classes if P ≠ NP:



Problem Classes if P = NP:



The 3SAT Problem

- Input: a sentence in propositional grammar, where each clause is a disjunction of 3 names which may be negated.
- Output: Either a mapping from names to values that satisfies the input sentence or **no way** (meaning there is no possible assignment that satisfies the input sentence)

3SAT Example

Sentence ::= *Clause*
Clause ::= $Clause_1 \vee Clause_2$ **(or)**
Clause ::= $Clause_1 \wedge Clause_2$ **(and)**
Clause ::= ¬*Clause* **(not)**
Clause ::= (*Clause*)
Clause ::= *Name*

$$\begin{aligned} \text{3SAT} (\quad & (a \vee b \vee \neg c) \\ & \wedge (\neg a \vee \neg b \vee d) \\ & \wedge (\neg a \vee b \vee \neg d) \\ & \wedge (b \vee \neg c \vee d)\ ) \end{aligned}$$

$$\rightarrow \{\ a\text{: }\textbf{true},\ b\text{: }\textbf{false},\ c\text{: }\textbf{false},\ d\text{: }\textbf{false}\}$$

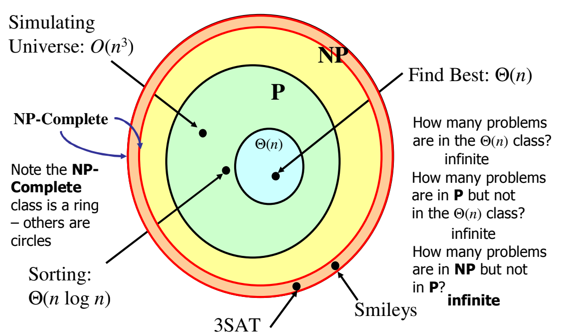
3SAT → Smiley

- Like 3/stone/apple/tower puzzle, we can convert every 3SAT problem into a Smiley Puzzle problem!
- Transformation is more complicated, but still polynomial time.
- So, if we have a fast (P) solution to Smiley Puzzle, we have a fast solution to 3SAT also!

NP Complete

- Cook and Levin proved that 3SAT was NP-Complete (1971) (Take CS660 to see how)
- A problem is NP-complete if it is as hard as the hardest problem in NP
- If 3SAT can be transformed into a different problem in polynomial time, than that problem must also be **NP-complete.**
- **Either all NP-complete problems are tractable (in P) or none of them are!**

Problem Classes if P ≠ NP:



Quiz Answers

- What would we need to do to prove a problem is $O(n^4)$?

  Find a procedure that solves the problem that is $O(n^4)$.

- What would we need to do to prove a problem is $\Omega(n^4)$?

  Prove that there is **no** procedure that solves the problem that is faster than $\Theta(n^4)$.

Quiz Responses: What is Computer Science?

Remembered (or almost remembered first class):

Study of "how to" knowledge, or whatever the fancy name for that is, The imperative study of procedures and language (I'm trying to remember what that definition was on the first day), "How to" do things; instead of declarative statements, talk about how to do things, I know we hade a specific definition in the beginning of the year but I forget it…I would say that computer science is a science/study of methods of manipulating data and information, the study of imperative knowledge; It doesn't need to have a computer and it's not a science, it doesn't deal with real things, instead numbers and data manipulations and problem solving; a liberal art, an innovative perspective on information; Study of imperative knowledge. Study of different types of problems and how to solve them. It's more of a liberal arts major than engineering.; liberal art! Learning about languages.; CS is not computer engineering, and its not science. Nevertheless, CS used sciences to prove computer problems.

Quiz Responses: What is Computer Science?

Problem solving (logic, language):

A way of thinking about computers and programs. A systematic way of creating and understanding how programs work, thinking differently, logically to how the computer language functions, It seems to be to theorize on how to systematically solve problems in a functional language, the study of logic and programs that can be understood by computers; Study of logics and way to solve problems; The study of using systems and algorithms to solve problems.; study of languages used to create intelligent computer procedures to solve usually huge or complex problems involving data ?!; the science of computers? Solving problems with computers; The study and use of languages to improve technology and program computers; CS is the art of getting a computer to do what you want through a common language.

Quiz Responses: What is Computer Science?

Amusing:
Computer Science is a great way to find new world.

The bane of my existence.

Computer Science

- (Expanded) Definition from Class 1:
  - Study of information processes
    - How to describe information processes by defining procedures
    - How to predict properties about information processes
    - How to elegantly and efficiently implement information processes in hardware and software

What have we spent most of our time on so far?

Where we've been, Where we're going

Computer Science: CS150 so far

- How to describe information processes by defining procedures
  - Programming with procedures, lists, recursion
  - Class 2, 4, 5, 6, 7, 8, 9, 10, 11, 12
- How to predict properties about information processes
  - Measuring work, Θ, O, Ω, complexity classes
  - Class 9, 10, 11, 12, 13, 14, 15, 16
- How to elegantly and efficiently implement information processes in hardware and software
  - Class 3 (rules of evaluation)

CS150 upcoming

- How to describe information processes by defining procedures
  - Programming with mutation, objects, databases, networks
- How to predict properties about information processes
  - What are we counting when we measure work?
  - Are there problems which can't be solved by procedures?
- How to elegantly and efficiently implement information processes in hardware and software
  - How to implement a Scheme interpreter
  - Not much in CS150 on hardware (see CS230 and CS333)

Famous Computer Scientists

- Ada – first computer scientist
  - She's in the course name!
- Grace Hopper – first compiler
  - First "bug", Navy ship, David Letterman nano
- John Backus – BNF, Fortran
  - UVa dropout
- Tony Hoare – Quicksort

Famous Computer Scientists

- Bill Gates (?)
  - Didn't invent Windows interface, word processor, PC, etc. (mostly invented by Doug Englebart and XEROX Parc)
  - Business notion that people would pay for software for PCs
  - Implemented a BASIC interpreter (but didn't invent BASIC)

NP Complete Problems

NP-Complete Problems

- Easy way to solve by trying all possible guesses
- If given the "yes" answer, quick (in P) way to check if it is right
  - Solution to puzzle (see if it looks right)
  - Assignments of values to names (evaluate logical proposition in linear time)
- If given the "no" answer, no quick way to check if it is right
  - No solution (can't tell there isn't one)
  - No way (can't tell there isn't one)

Traveling Salesperson Problem

- Input: a graph of cities and roads with distance connecting them and a minimum total distant
- Output: either a path that visits each with a cost less than the minimum, or "no".

- If given a path, easy to check if it visits every city with less than minimum distance traveled

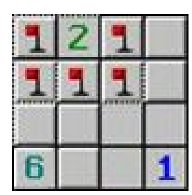
Graph (Map) Coloring Problem

- Input: a graph of nodes with edges connecting them and a minimum number of colors
- Output: either a coloring of the nodes such that no connected nodes have the same color, or "no".

If given a coloring, easy to check if it no connected nodes have the same color, and the number of colors used.

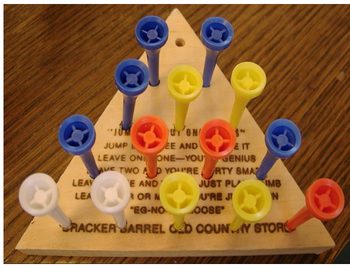
Minesweeper Consistency Problem

- Input: a position of n squares in the game Minesweeper
- Output: either a assignment of bombs to squares, or "no".



- If given a bomb assignment, easy to check if it is consistent.

Pegboard Problem



Pegboard Problem

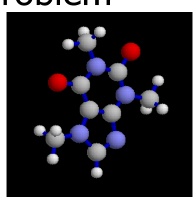
- Input: a configuration of n pegs on a cracker barrel style pegboard
- Output: if there is a sequence of jumps that leaves a single peg, output that sequence of jumps. Otherwise, output **false**.

If given the sequence of jumps, easy ($O(n)$) to check it is correct. If not, hard to know if there is a solution.

Drug Discovery Problem

– Input: a set of proteins, a desired 3D shape

– Output: a sequence of proteins that produces the shape (or impossible)



Caffeine

If given a sequence, easy (not really) to check if sequence has the right shape.

Note: US Drug sales = $200B/year

Is it ever *useful* to be confident that a problem is hard?

Hint: PS4

Charge

- AC's exam review is tonight at 7pm
- Friday:
  - Exam 1 out (will be posted on web also)
  - How Lorenz was really broken