Motivations for Complexity Theories

If we can call B's solution to solve problem A, then A is ___ than B.

NP ≠ co-NP implies P ≠ NP

Motivation

- Question: Complexity theories are abstract and complex/subtle. Why learning it?
- Answer:
  - Train your brain to work in the advanced abstract world
    - A stepping stone to the Theoretical Computer Science (TCS) and in general (modern) science
    - Esp. important in the CURRENT AI era
  - Has many practical implications in problem solving & algorithm design
    - No need to attempt to design an algorithm to solve a NPC/NP-hard problem

Ruler-and-Compass

- Ancient Greek mathematicians developed the methodology of “ruler-and-compass” constructions:
  - if one is given only a ruler (without marks) and a compass, what objects can be constructed as a result of a finite set of operations?
  - While they achieved many successes, three problems confounded their efforts: (1) squaring the circle; (2) trisecting an angle; and (3) duplicating a cube (i.e., constructing a cube whose volume is twice that of a given cube).

https://mathscholar.org/2018/09/simple-proofs-the-impossibility-of-trisection/

Impossibility of trisecting a 60 degree angle

- Q: Why proving "impossibility" is hard?
- A: How do you show someone much smarter/luckier cannot do it?

Gist of an elementary Proof

- "We will show that trisecting a 60 degree angle, in particular, is equivalent to constructing the number cos(π/9), which is an algebraic number that satisfies an irreducible polynomial of degree 3."
- "Since the only numbers and number fields that can be produced by ruler-and-compass construction have algebraic degrees that are powers of two, this shows that the trisection of a 60-degree angle is impossible. "

Informally & non-rigorously

$$1/2 = \cos(\frac{\pi}{3}) = 4\cos^3(\frac{\pi}{9}) - 3\cos(\frac{\pi}{9})$$

$$8x^3 - 6x - 1 = 0$$

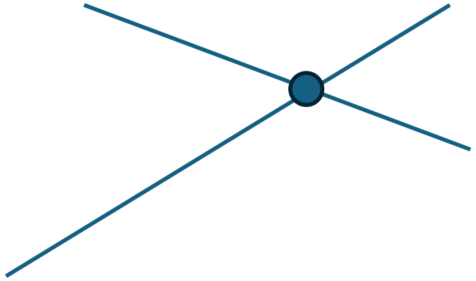
$$x = \sqrt[3]{\frac{1+\sqrt{3}i}{16}} + \sqrt[3]{\frac{1-\sqrt{3}i}{16}}$$

Ruler-&-compass can only get numbers computed by +, -, *, /, √

Hence, cannot get the above x or $\sqrt[3]{2}$

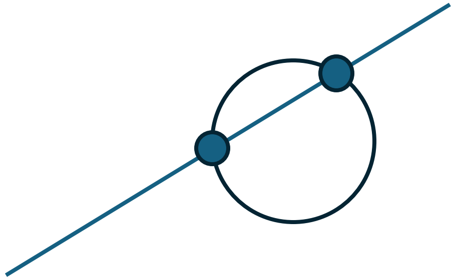
https://mathscholar.org/2018/09/simple-proofs-the-impossibility-of-trisection/

Abstraction of Ruler-and-Compass's Capability: 3 cases



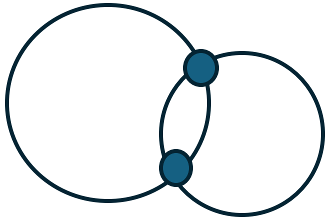
$$ax + b = cx + d$$

x is a rational number



$$(x, cx + d) \in (x-a)^2 + (y-b)^2 = e$$

x involve at most √ (and its nesting)



$$(x-a)^2 + (y-b)^2 = e \cap (x-c)^2 + (y-d)^2 = f$$

x involve at most √ (and its nesting)

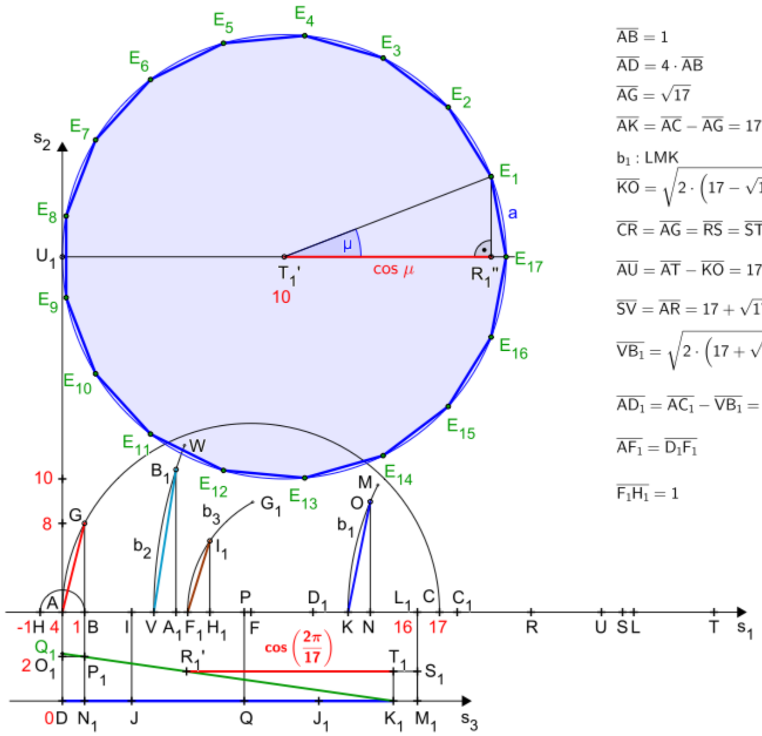
Gauss's Proof of (the Existence of Constructing) a regular 17-gon

> "that constructibility is equivalent to expressibility of the trigonometric functions of the common angle in terms of arithmetic operations and square root extractions …"

Ingenious | Gauss, 1796

$$\cos\frac{2\pi}{17} = \frac{1}{16}\left(\sqrt{17} - 1 + \sqrt{34 - 2\sqrt{17}}\right) + \frac{1}{8}\left(\sqrt{17 + 3\sqrt{17} - \sqrt{34 - 2\sqrt{17}} - 2\sqrt{34 + 2\sqrt{17}}}\right)$$

Sophisticated



$\overline{AB} = 1$

$\overline{AC} = 17 \cdot \overline{AB}$

$\overline{AD} = 4 \cdot \overline{AB}$

$\overline{AF} = \overline{FC}$

$\overline{AG} = \sqrt{17}$

$\overline{AI} = \overline{DJ} = -1 + \sqrt{17}$

$\overline{AK} = \overline{AC} - \overline{AG} = 17 - \sqrt{17}$

$\overline{AL} = 2 \cdot \overline{AK} = 2 \cdot \left(17 - \sqrt{17}\right)$

$b_1 : LMK$

$\overline{KN} = 1$

$\overline{KO} = \sqrt{2 \cdot \left(17 - \sqrt{17}\right)}$

$\overline{AP} = \overline{DQ} = \overline{AI} + \overline{KO} = -1 + \sqrt{17} + \sqrt{2 \cdot \left(17 - \sqrt{17}\right)}$

$\overline{CR} = \overline{AG} = \overline{RS} = \overline{ST}$

$\overline{AT} = \overline{AC} + 3 \cdot \overline{AG} = 17 + 3 \cdot \sqrt{17}$

$\overline{AU} = \overline{AT} - \overline{KO} = 17 + 3 \cdot \sqrt{17} - \sqrt{2 \cdot \left(17 - \sqrt{17}\right)}$

$\overline{SV} = \overline{AR} = 17 + \sqrt{17}$

$b_2 : SWV; \quad \overline{VA_1} = 1$

$\overline{VB_1} = \sqrt{2 \cdot \left(17 + \sqrt{17}\right)}$

$\overline{AC_1} = \overline{AU} - \overline{VB_1} = 17 + 3 \cdot \sqrt{17} - \sqrt{2 \cdot \left(17 - \sqrt{17}\right)} - \sqrt{2 \cdot \left(17 + \sqrt{17}\right)}$

$\overline{AD_1} = \overline{AC_1} - \overline{VB_1} = 17 + 3 \cdot \sqrt{17} - \sqrt{2 \cdot \left(17 - \sqrt{17}\right)} - 2 \cdot \sqrt{2 \cdot \left(17 + \sqrt{17}\right)}$

$\overline{AF_1} = \overline{D_1F_1}$

$b_3 : D_1G_1F_1$

$\overline{F_1H_1} = 1$

$\overline{F_1I_1} = \sqrt{17 + 3 \cdot \sqrt{17} - \sqrt{2 \cdot \left(17 - \sqrt{17}\right)} - 2 \cdot \sqrt{2 \cdot \left(17 + \sqrt{17}\right)}}$

$\overline{QJ_1} = \overline{F_1I_1} = \overline{J_1K_1}$

$\overline{AL_1} = \overline{DM_1} = 16 \cdot \overline{AB}$

$\overline{DO_1} = \overline{O_1A} = 2 = \overline{N_1P_1}$

$\overline{N_1P_1} : \overline{N_1M_1} = 2 : 15 = \overline{Q_1O_1} : 1 = 0,1\overline{3} = 0,4 : 3$

$10 \cdot 0,1\overline{3} = 1,\overline{3} = \frac{4}{3} \Rightarrow \frac{1}{3} \cdot \overline{L_1M_1} = \overline{M_1S_1} = 1,\overline{3}$

$\overline{S_1R_1'} \parallel \overline{DM_1}$; U_1 : position on s2 is arbitrary; $\overline{U_1T_1'} = 10 \cdot \overline{AB}$

$$\cos\left(\frac{2\pi}{17}\right) = \frac{1}{16}\left(-1 + \sqrt{17} + \sqrt{2 \cdot \left(17 - \sqrt{17}\right)} + 2 \cdot \sqrt{17 + 3 \cdot \sqrt{17} - \sqrt{2 \cdot \left(17 - \sqrt{17}\right)} - 2 \cdot \sqrt{2 \cdot \left(17 + \sqrt{17}\right)}}\right)$$

Just labor work | Richmond, 1893

Almost 100 years afterwards

https://en.wikipedia.org/wiki/Heptadecagon

Algorithmic Problem

- Is there an efficient algorithm to determine if a graph has a clique of size k? k-Clique Problem

1. Solvable by enumerating all size-k subset of vertices ➔ $O\left(\binom{n}{k}\right)$
2. Can we do better than that complexity? ➔ NO! (in the worst case)

Gist of the informal Proof

- There is no "efficient(1)" algorithm to solve 3-SAT
- Any 3-SAT instance can be transformed "efficiently(2)" to an "equivalent" instance of k-Clique
  - Answers of the two problems are Yes-to-yes & No-to-no
- Hence, if k-Clique can be solved efficiently, so does 3-SAT ➔ contradition!

Karp's reduction ➔ Polynomial time reduction in our slides https://en.wikipedia.org/wiki/Halting_problem

Even “harder” Problems

Undecidable

Halting Problem

- Given **any** program P and **any** input x, can a super-debugger quickly tell me whether P(x) will eventually finish or run forever?

NO!

No algorithm can always give the right yes/no answer for every program/input pair
-- Alan Turing, 1936

Proof by contradiction:

```
HALT?(P, x)   # always halts, always right
```

```
DIAGONAL(Q):
  if HALT?(Q, Q) == YES:   # does Q halt on its *own* code?
    loop forever           # contradict the prediction
  else:                    # HALT? said Q doesn't halt
    halt immediately       # contradict again!
```

DIAGONAL(DIAGONAL)

`HALT?` **predicts**	**inside** `DIAGONAL` **we**	**reality**	**contradiction?**
“YES” (will halt)	loop forever	doesn’t halt	**yes**
“NO” (won’t)	halt immediately	halts	**yes**

Russell's paradox
Self-referencing