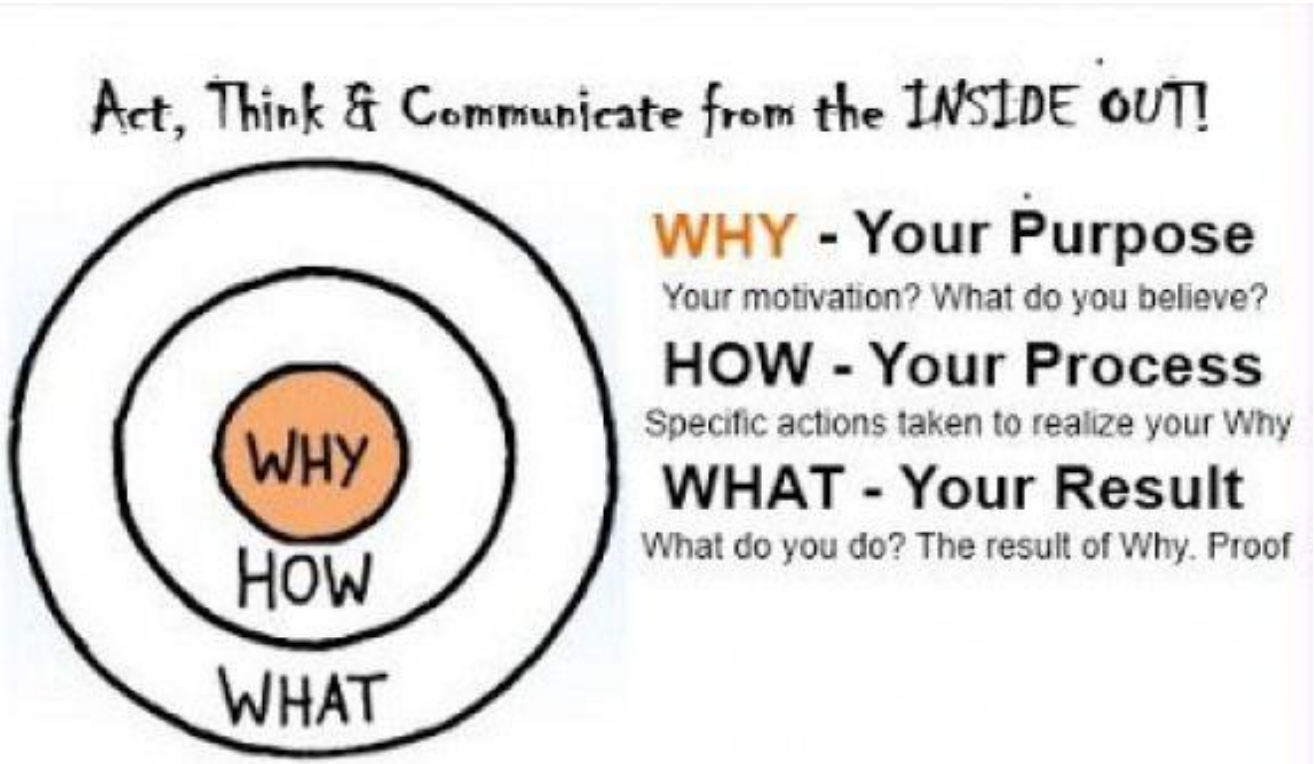
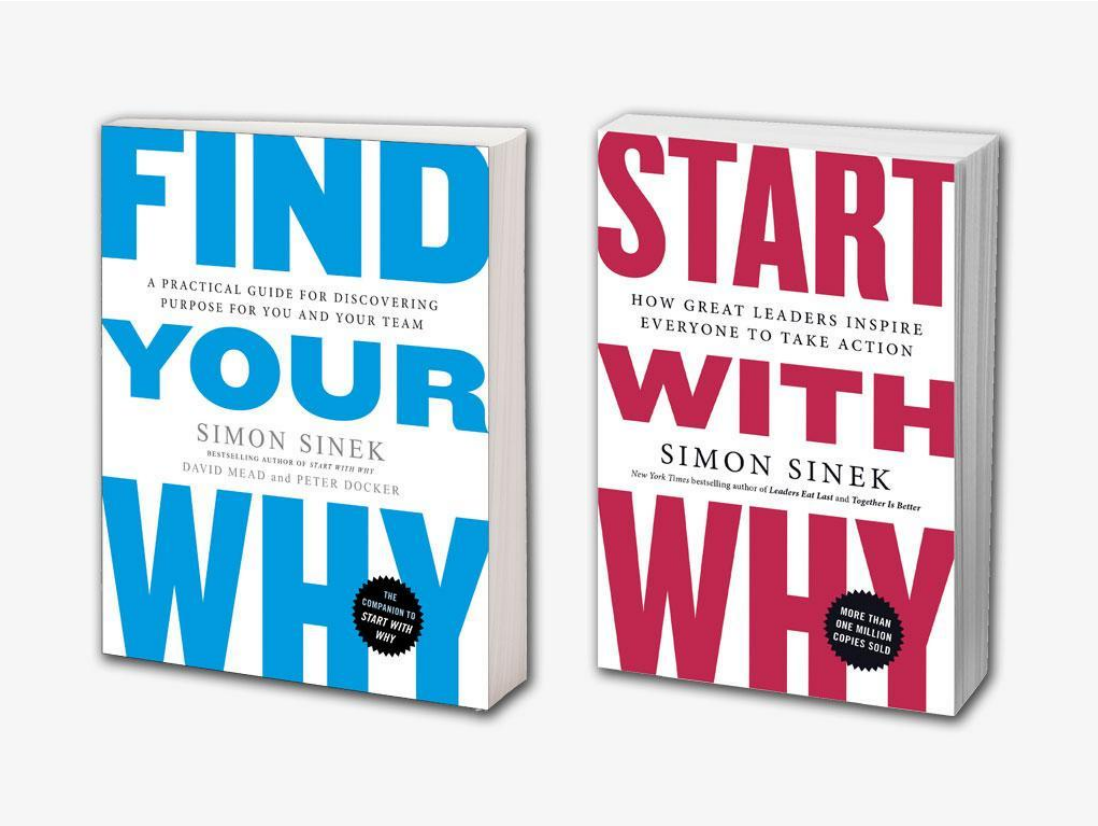
Theory of Computation

Slides 1

Dr. Talha Waheed
UET, Lahore

Theory of Computing – Why, How and What?





Goals

1. Core of everything - To introduce elegant theory that underlies computing since its early days (and it is still alive).

2. To motivate students about (abstract) models of computation and their limits

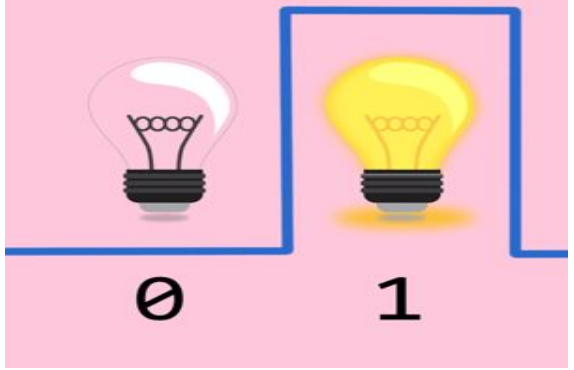
3. To show students how to apply theory in their own work (applications).

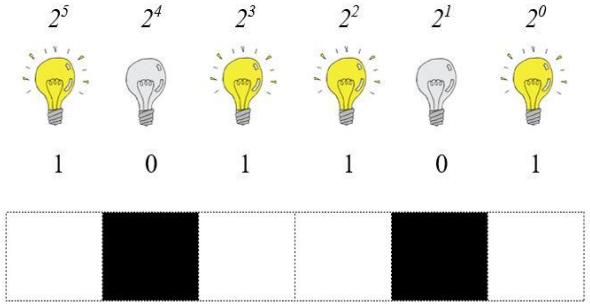
Why do we study Computing?

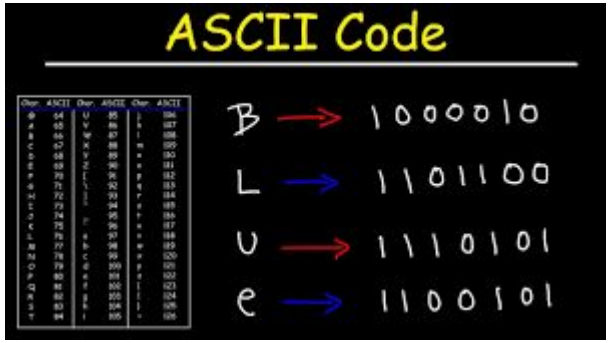
- What is Computer Science? What and how do we do?
- Is it a right name for the discipline, what else you suggest?
- What is computing? What other names do you suggest?
- Is computing depends on the medium?
- Are they only electronic or digital?
- syntax is independent of medium (not intrinsic to physics).
- We assign semantics to syntax (symbols)

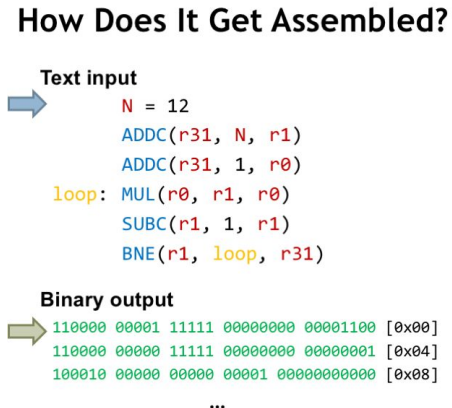
Syntax

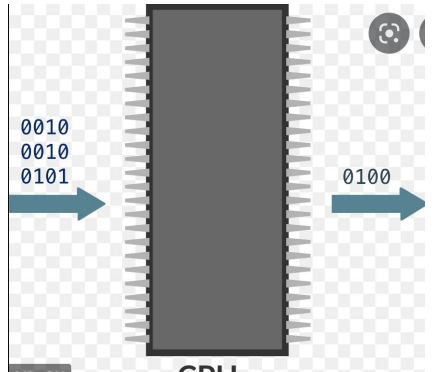
- Programs are purely syntax (formal rules).
- Is Data & Programs (in Computers) are defined syntactically by Binary numbers?
- Is Syntax a physical feature of computers?
  - **Syntax is not intrinsic (built in) to physics.**
- Computational states (syntax) are not built-in within the physical medium, they are assigned by some outside person.
- Hardware realization to computational description is abstract.
- A water pump/birds/mechanical computer or even "a group of pigeons can be trained to peck as a Turing machine”





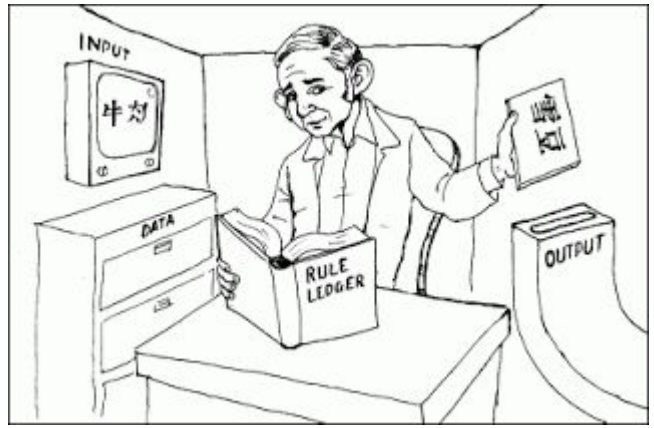






Semantics

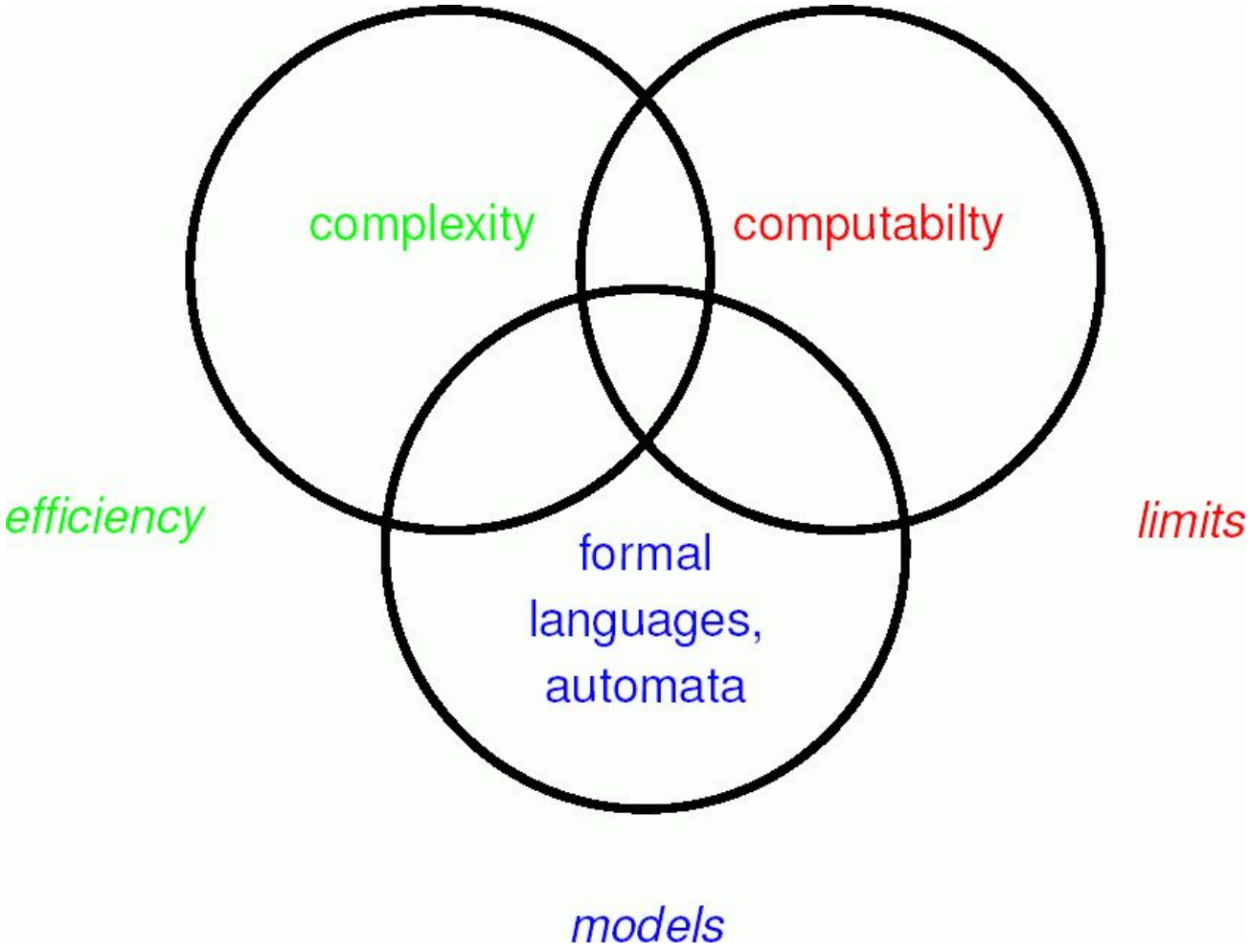
- Next issueis : How do the syntax get its meaning (semantics)?
    - **Even Semantics is not intrinsic (built-in) to syntax.**
- An outside person encodes some information in a form that can be processed by the circuitry of the computer. He/she provides a syntactical realization of the information that the computer can implement in, e.g., different voltage levels.



Chinese Room Argument

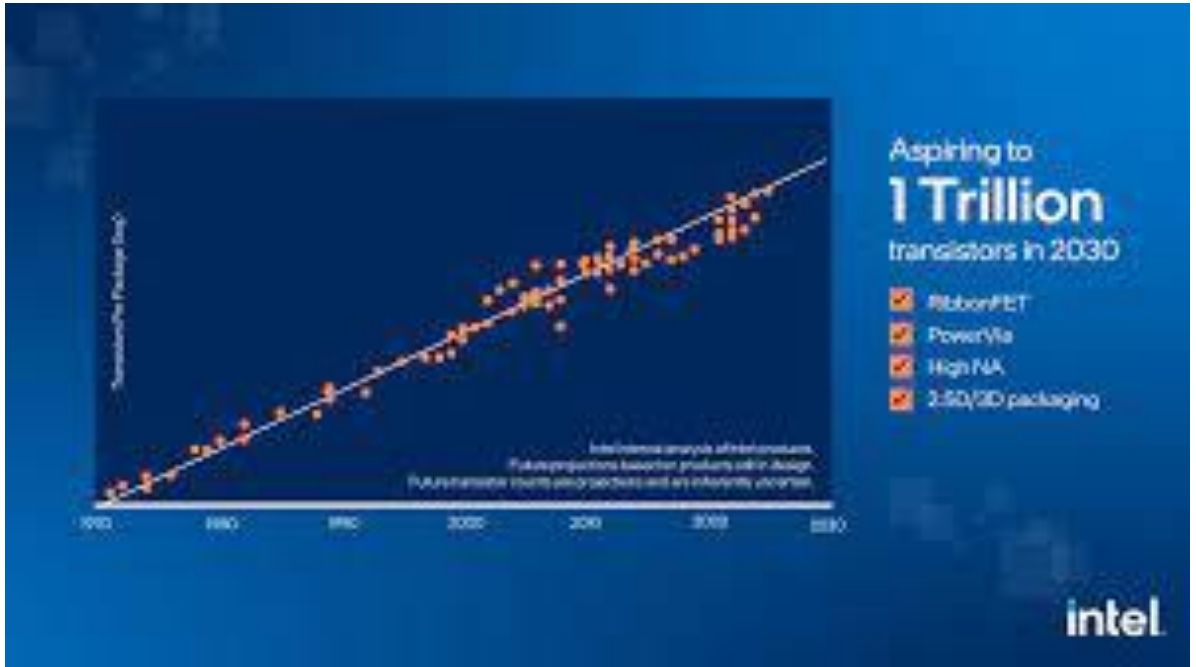
- Computer goes through a series of electrical signals that the outside person can interpret both syntactically and semantically even though,
  of course, the hardware has no intrinsic syntax or semantics:
    - **It is all in the eye of the beholder.**

Three Interlinked Subjects



Moore’s Law of Transistor in the Computer

- Moore stated it in 1965
- **speed and capability of computers can be expected to double every two years**, as a result of increases in the number of transistors a microchip can contain.

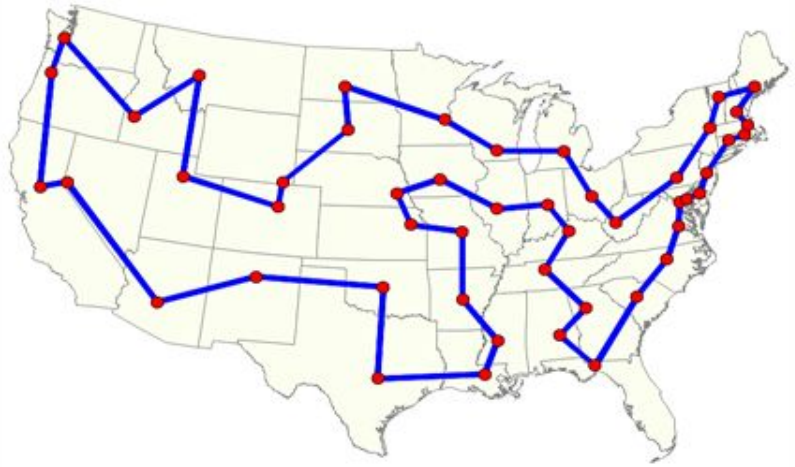


How big is big?

- **Earth has 7.5 x 10^{18} grains** of sand
- Different estimates 10^{24} **or 200 billion trillion stars**
- 70 thousand million, million, million stars in the observable universe (a 2003 estimate), so that we've got **multiple stars for every grain of sand.**

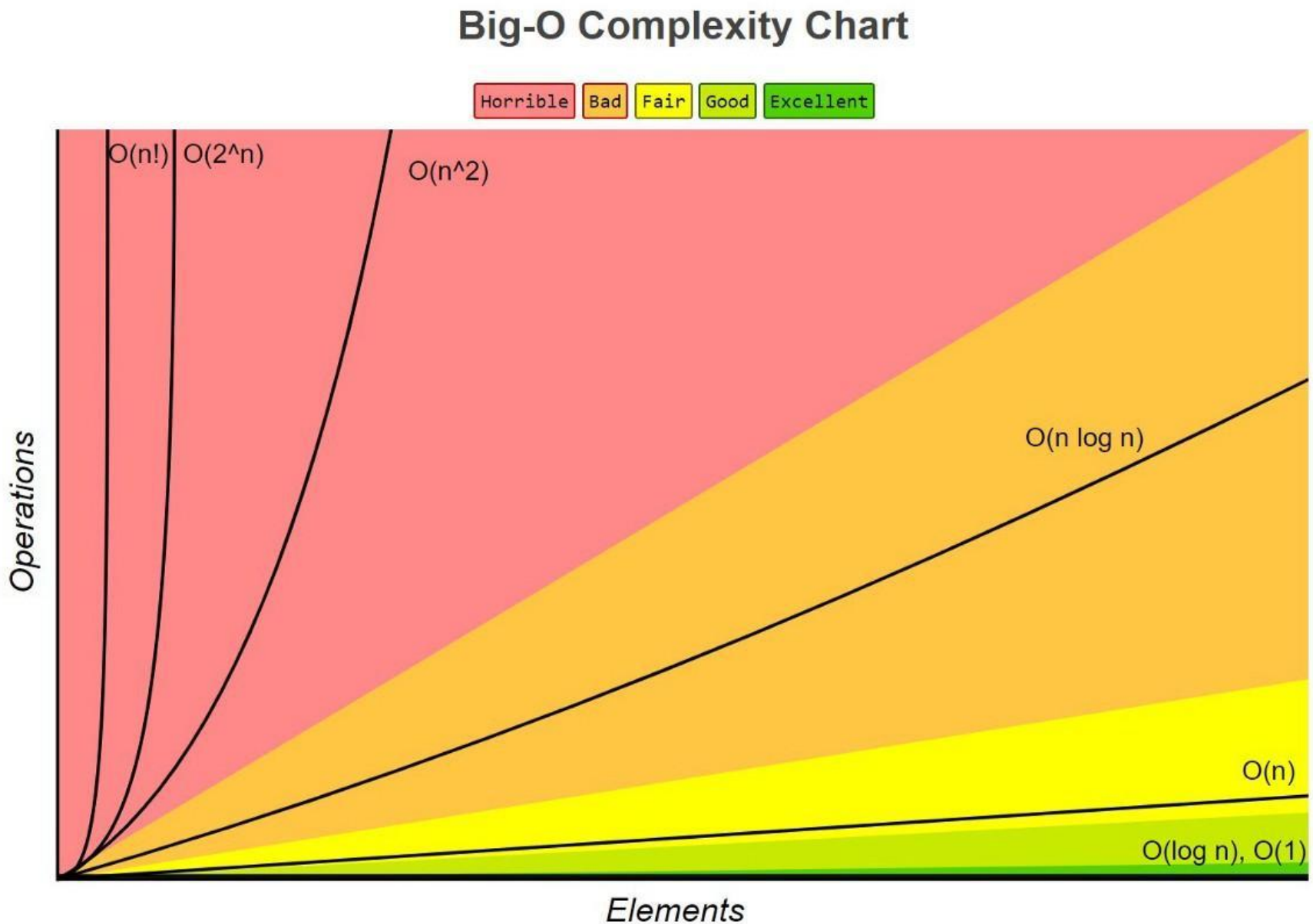
- 10^{40} **Possibilities in Checker (Chinese Chess)**
- 10^{120} **Possibilities in Chess**
- 10^{-350} **Probability of Human Evolution from Single cell**
- 10^{-950} **Probability of Human Evolution (if we consider Genes)**

Travelling Salesman problem



n	n!
0	1
1	1
2	2
3	6
4	24
5	120
6	720
7	5040
8	40320
9	362880
10	3628800
11	39916800
12	479001600

- The speed of light is $3*10^{8}$ m/sec. The width of a proton is 10^{-15} m. So, if we perform one operation in the time it takes light to cross a proton, we can perform $3*10^{23}$ operations/sec. There have been about $3*10^{17}$ seconds since the Big Bang.
- So, at that rate, we could have performed about $9*10^{40}$ operations since the Big Bang. But 36! is $3.6*10^{41}$.
- So there hasn't been enough time since the Big Bang to have solved even a single traveling salesman problem with 37 cities. That's fewer than one city per state in the United States.
- $43! = 6*10^{52}$
- $70! = 1.2 *10^{100}$
- Google?
- Googleplex?


Big-O Complexity Chart
Horrible
Bad
Fair
Good
Excellent
O(n!)
O(2^n)
O(n^2)
O(n log n)
O(n)
O(log n), O(1)
Operations
Elements

Tractable problems

A **tractable problem** is one that is said to be solvable in a **polynomial** (reasonable) time. In simple terms this means that the algorithm that solves the problem runs quickly enough for it to be practical on a computer.

Mathematical notation	Name	Tractable / Intractable
$n!$	*Exponential time*	*Intractable*
2^n	*Exponential time*	*Intractable*
n^2	*Polynomial time*	*Tractable*
n	*Linear time*	*Tractable*
$\log n$	*Logarithmic time*	*Tractable*