1 | sources

1.1 | gentle introductions

1.1.1 | `https://en.wikipedia.org/wiki/Computational_complexity_theory`

1.1.2 | `https://complexityzoo.net/Petting_Zoo`

2 | overview

2.1 | computational complexity theory studies how ”difficult” a problem is

2.1.1 | importantly, not how ”good” an algorithm is... this field deals with all algorithms that solve a given problem

2.2 | key concepts

2.2.1 | types of problems

2.2.2 | Turing machines

2.2.3 | reducibility

2.2.4 | complexity classes

2.2.5 | hierarchy

2.3 | key problems

2.3.1 | P vs NP

3 | flows

3.1 | Wikipedia computational complexity theory

3.1.1 | computational problems

1. problem instances

   A problem describes the problem. the actual ”numbers” that describe a specific problem is called a problem instance. sorting a list is a problem, sorting *this* list is a problem instance.

2. representing problem instances

   formally strings of characters from alphabets.