

# P problem 和 NP problem

---

1. P问题: 可以在多项式时间内解决的问题。
2. NP问题: 可以在多项式时间内验证一个解是否正确的问题。
3. NPC问题:

$NP$ 问题: 可以在多项式时间内猜出一个解的问题。很显然,  $NP$ 问题难于P问题。

## 世界著名难题: $NP=P?$ 问题

很容易看到, 所有的P问题都是NP问题。即  $P \subseteq NP$ 。现在, 我们关心是否有  $NP=P$ 。如果成立的话, 所有NP问题, 都存在多项式时间解法。或者说, 所有的NP问题都是转换为P问题。

## 规约

问题A可以规约为问题B, 即可以用B的解法来解决A问题。或者说A可以通过B解决, 或者说A可以转换为B问题。比如, A问题: 求解一元一次方程。B问题: 求一元二次方程。很显然A问题可以规约为B问题, 而且我们可以看到A可以规约为B问题, 说明B的时间复杂度大于等于A的时间复杂度。这是因为, 如果B的时间复杂度比A小, 那么A可以通过B来解决, 则A的算法可以改进为B的算法, 此时两者的时间复杂度相同。

A和B的规约有一个限制, 即两者的转换规则是在多项式时间内完成的。(多项式规约, 非多项式规约没意义)

规约具有传递性。这一点很显然。

标准概念: 如果可以找到一个转换规则, 对任意一个A的输入, 都可以按照这个规则转换为B的输入, 使得两个程序的输出相同, 那么A可以规约为B。

## 引入NPC问题

现在, 我们希望有这么一个问题, 即所有的NP问题都可以规约为一个超级问题。此问题就是NPC问题。详细来说, 就是我们可以规约一个问题到另一个问题, 虽然时间复杂度增加了, 但是规约后的问题应用范围增加了。也就是说我们可以通过不断的规约, 来寻找时间复杂度越来越高, 应用范围越来越广的问题。

显然, 我们可以通过解决该超级问题, 即NPC问题, 来解决所有NP问题。

## 第一个NPC问题

该问题是一个逻辑电路问题(可以搜索), 该问题和SAT问题本质上是一样的。