

贪心算法

思路

- 可能要先对数据进行排序
- 证明思路
 - 反证法=。=
 - **The Greedy Algorithm Always Stays Ahead:** Show that after each step of the greedy algorithm, its solution is at least as good as any other algorithm's.
 - **Structural:** Discover a simple "structural" bound asserting that every possible solution must have a certain value. Then show that your algorithm always achieves this bound.
 - **Exchange argument:** Gradually transform any solution to the one found by the greedy algorithm without hurting its quality.

典型问题

1. Interval scheduling

问题描述: 一组课程, 有开始和结束时间, 找出时间安排不会重叠的最大课程数目。

解决方法: 对课程结束时间从早到晚进行排序。

2. Interval Partitioning

问题描述: 一组课程, 有开始和结束时间, 找出把它们全部安排下的最小教室数目。

解决方法: 对课程开始时间从早到晚进行排序。

3. Scheduling to minimize lateness

问题描述: 一组任务, 有完成任务需要花费的时间, 以及 deadline, 给出一种安排方案, 使得每个任务延期的时间之和最小。

解决方法: 对 deadline 从早到晚进行排序。

4. k-聚类

问题描述: 把许多点分成 k 类, 使得每个类别间的距离最大。

解决方法: TODO

其它问题

- Dijkstra 算法解决无负权重的最短路问题;
- Prim, Kruskal 解决最小生成树问题;
-