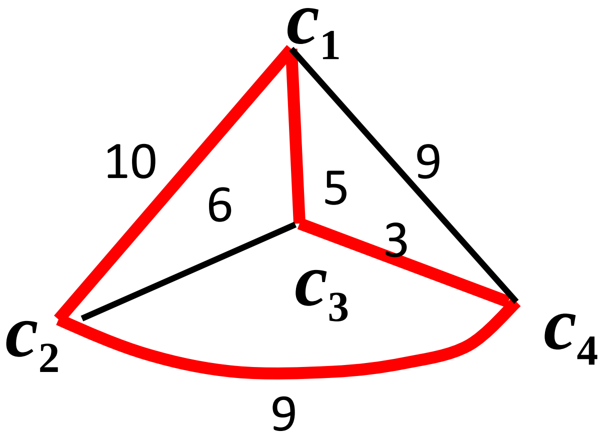
货郎问题与计算复杂性理论

例4 货郎问题

问题：

有n个城市，已知任两个城市之间的距离. 求一条每个城市恰好经过1次的回路，使得总长度最小.



建模与算法

- 输入

  有穷个城市的集合 $C = \{ c_1, c_2, \ldots, c_n\}$,

  距离 $d(c_i,c_j) = d(c_j,c_i) \in Z^+$, $1 \le i < j \le n$

- 解：$1, 2 \ldots, n$ 的排列 $k_1, k_2, \ldots, k_n$ 使得：

$$\boxed{\min\{\sum_{i=1}^{n-1} d(c_{k_i}, c_{k_{i+1}}) + d(c_{k_n}, c_{k_1})\}}$$

- 现状：至今没找到有效的算法

0−1背包问题

0-1背包问题：有n个件物品要装入背包，第i 件物品的重量 w_i, 价值v_i，i=1,2,…,n. 背包最多允许装入的重量为B, 问如何选择装入背包的物品，使得总价值达到最大?

实例：n=4，B=6，物品的重量和价值如下

标号	1	2	3	4
重量 w_i	3	4	5	2
价值 v_i	7	9	9	2

0-1背包问题建模

问题的**解**：**0-1向量** $< x_1, x_2, ..., x_n >$

$x_i = 1 \Leftrightarrow$ 物品 i 装入背包

目标函数 $\max \sum_{i=1}^{n} v_i x_i$

约束条件 $\sum_{i=1}^{n} w_i x_i \le B$

$x_i = 0,1, i = 1,2,...,n$

双机调度

双机调度问题：有n项任务, 任务 i 的加工时间为 t_i , $t_i \in Z^+$, i=1,2,…,n. 用两台相同的机器加工，从0时刻开始计时，完成时间是后停止加工机器的停机时间. 问如何把这些任务分配到两台机器上，使得完成时间达到最小？

实例：任务集 S ={1,2,3,4,5,6}

$\underline{t_1=3,\ t_2=10},\ t_3=6,\ \underline{t_4=2},\ t_5=1,\ t_6=7$

解：机器 1 的任务：1, 2, 4

机器 2 的任务：3, 5, 6

完成时间： max{ 3+10+2, 6+1+7 }=15

双机调度建模

解: 0-1向量 $<x_1, x_2, ..., x_n>$, x_i=1表示任务 i 分 配到第一台机器，i =1,2,...,n.

不妨设机器1的加工时间 ≤ 机器2的加工时间令$T=t_1+t_2+...+t_n$，$D=\lfloor T/2 \rfloor$，机器1的加工时间不超过D，且达到最大.

如何对该问题建模?目标函数与约束条件是什么？

NP-hard问题

- 这样的问题有数千个，大量存在于各个应用领域.
- 至今没找到有效算法：现有的算法的运行时间是输入规模的指数或更高阶函数.
- 至今没有人能够证明对于这类问题不存在多项式时间的算法.
- 从是否存在多项式时间算法的角度看，这些问题彼此是等价的. 这些问题的难度处于可有效计算的边界.

Algorithm + Data Structure = Programming

好的算法

- 提高求解问题的效率
- 节省存储空间

算法的研究目标

问题→建模并寻找算法	算法设计技术
算法→算法的评价	算法分析方法
算法类→问题复杂度估计	问题复杂度分析
问题类→能够求解的边界	计算复杂性理论

课程主要内容

近似算法 | 随机算法

NP 完全理论简介

算法分析与问题的计算复杂性

分治策略 | 动态规划 | 贪心算法 | 回溯与分支限界

数学基础、数据结构

计算复杂性理论:
NP完全理论
其他算法

算法设计:
算法分析方法
算法设计技术
基础知识

算法研究的重要性

算法设计与分析技术在计算机科学与技术领域有着重要的应用背景

算法设计分析与计算复杂性理论研究是计算机科学技术的核心研究领域

- 1966-2005期间，Turing奖获奖50人, 其中10人以算法设计, 7人以计算理论、自动机和复杂性研究领域的杰出贡献获奖
- 计算复杂性理论的核心课题“P=NP?”是本世纪 7个最重要的数学问题之一

提高学生素质和分析问题解决问题的能力，培养计算思维

小结

- NP-hard问题的几个例子：货郎问题 0-1背包问题、双机调度问题等
- NP-hard问题的计算现状
- 计算复杂性理论的核心——NP完全理论
- 算法研究的主要内容及重要意义