

ACM 程序设计

计算机学院 刘春英

调课三周

(11/6, 11/13, 11/20)

今天，

你 AC 了吗？

每周一星（5）：



枫冰叶子

第六讲

贪心算法 (Greedy Algorithm)

还记得hdoj_1009吗？

FatMouse' Trade

所谓“贪心算法”是指：

在对问题求解时，总是作出在当前看来是最好的选择。也就是说，不从整体上加以考虑，它所作出的仅仅是在某种意义上的局部最优解（是否是全局最优，需要证明）。

特别说明：

若要用贪心算法求解某问题的整体最优解，必须首先证明贪心思想在该问题的应用结果就是最优解！！

用事实说话——

实例分析

一、事件序列问题

已知N个事件的发生时刻和结束时刻（见下表，表中事件已按结束时刻升序排序）。一些在时间上没有重叠的事件，可以构成一个事件序列，如事件{2, 8, 10}。事件序列包含的事件数目，称为该事件序列的长度。请编程找出一个最长的事件序列。

事件编号	0	1	2	3	4	5	6	7	8	9	10	11
发生时刻	1	3	0	3	2	5	6	4	10	8	15	15
结束时刻	3	4	7	8	9	10	12	14	15	18	19	20

算法分析：

- 不妨用 $\text{Begin}[i]$ 和 $\text{End}[i]$ 表示事件 i 的开始时刻和结束时刻。则原题的要求就是找一个最长的序列 $a_1 < a_2 < \dots < a_n$ ，满足：

$$\text{Begin}[a_1] < \text{End}[a_1] \leq \dots \leq \text{Begin}[a_n] < \text{End}[a_n]$$

可以证明，如果在可能的事件 $a_1 < a_2 < \dots < a_n$ 中选取在时间上不重叠的最长序列，那么一定存在一个包含 a_1 （结束最早）的最长序列。

（证明：略）

思考：

- 请谈谈自己的解题思路

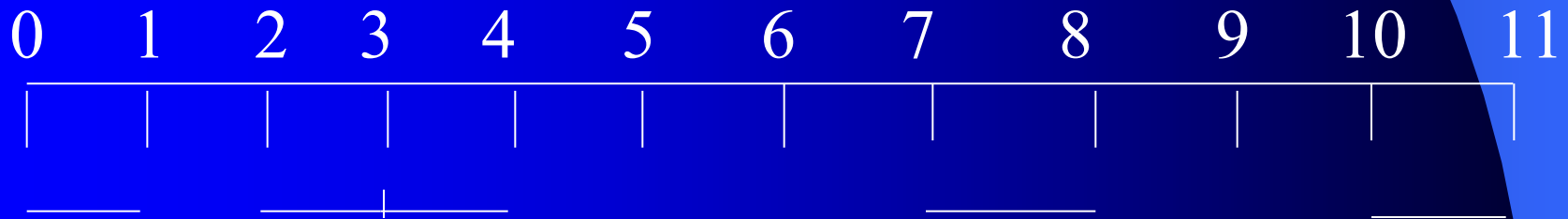
练习题目：

2037 今年暑假不AC

二、区间覆盖问题

用 i 来表示 x 轴上坐标为 $[i-1, i]$ 的区间（长度为1），并给出 M ($1 \leq M \leq 200$) 个不同的整数，表示 M 个这样的区间。现在让你画几条线段覆盖住所有的区间，条件是：每条线段可以任意长，但是要求所画线段之和最小，并且线段的数目不超过 N ($1 \leq N \leq 50$)。

例如： $M=5$ 个整数1、3、4、8和11表示区间，要求所用线段不超过 $N=3$ 条



算法分析：

- 如果 $N \geq M$ ，那么显然用 M 条长度为1的线段可以覆盖住所有的区间，所求的线段总长为 M 。
- 如果 $N=1$ ，那么显然所需线段总长为：...
- 如果 $N=2$ ，相当于 $N=1$ 的情况下从某处断开（从哪儿断开呢？）。
- 如果 $N=k$ 呢？

三、HDOJ_1050 Moving Tables

- [题目链接](#)

Sample Input

```
3
4
10 20
30 40
50 60
70 80
2
1 3
2 200
3
10 100
20 80
30 50
```

Sample Output

```
10
20
30
```

算法分析：

- 1、如果没有交叉，总时间应该是多少？
- 2、影响搬运时间的因素是什么？
- 3、如果每趟处理都包含最大重叠，处理后的效果是什么？
- 4、得出什么结论？

HDOJ_1050源码：

```
#include <iostream>
using namespace std;
int main()
{ int t,i,j,N,P[200];
  int s,d,temp,k,min;
  cin>>t;
  for(i=0;i<t;i++)
  {
    for(j=0;j<200;j++)
      P[j]=0;
    cin>>N;
    for(j=0;j<N;j++)
    {
      cin>>s>>d;
      s=(s-1)/2;
      d=(d-1)/2;
```

```
      if(s>d)
      { temp=s;
        s=d;
        d=temp; }
      for(k=s;k<=d;k++)
        P[k]++;
    }
    min=-1;
    for(j=0;j<200;j++)
      if(P[j]>min)
        min=P[j];
    cout<<min*10<<endl;
  }
  return 0;
}
```

贪心算法的基本步骤

- 1、从问题的某个初始解出发。
- 2、采用循环语句，当可以向求解目标前进一步时，就根据局部最优策略，得到一个部分解，缩小问题的范围或规模。
- 3、将所有部分解综合起来，得到问题的最终解。

贪心算法都很简单吗？

看一道难一些的。

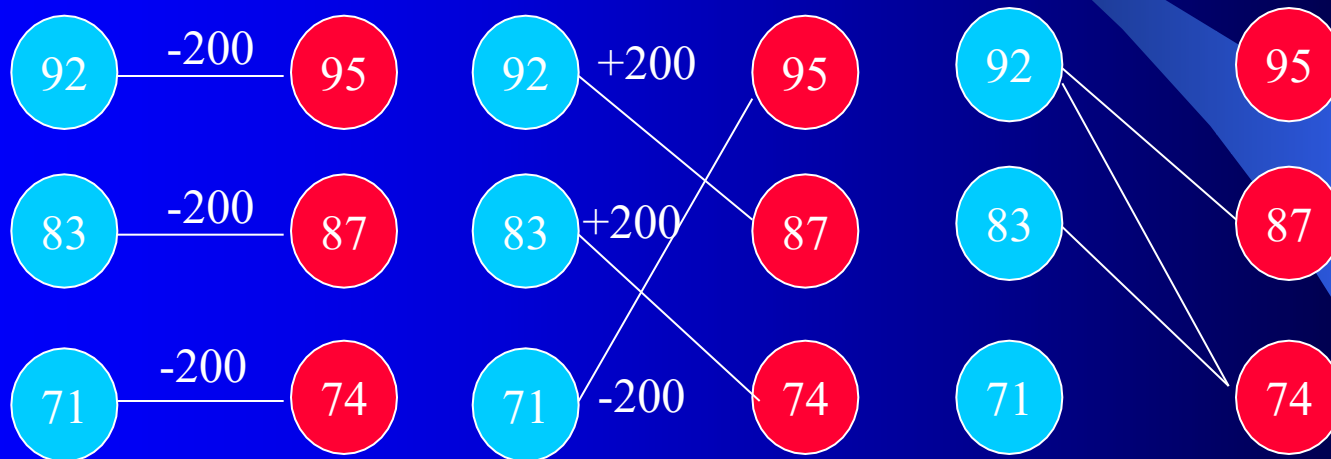
(2004年上海赛区试题：当时算是简单题)

ACM-ICPC Asia Regional,
2004, Shanghai

Problem H

Tian Ji—The Horse Racing

示意图:



谈谈自己的想法——

Case 1:

King:	200	180	160
Tianji:	190	170	150

Case 2:

King:	200	180	160
Tianji:	180	170	150

Case 3:

King:	200	180	160
Tianji:	180	155	150

总体的思路 是什么？

提醒：

很多贪心类型的题目都象本题一样，不是最朴素的贪心，而是需要做一些变化，对于我们，关键是找到贪心的本质！

本讲重点：

(连通网的)最小生成树

问题：

假设要在 n 个城市之间建立通讯联络网，则连通 n 个城市只需要修建 $n-1$ 条线路，如何在最节省经费的前提下建立这个通讯网？

该问题等价于：

构造网的一棵最小生成树，即：
在 e 条带权的边中选取 $n-1$ 条边（不构成回路），使“**权值之和**”为最小。

算法一：（普里姆算法）

算法二：（克鲁斯卡尔算法）

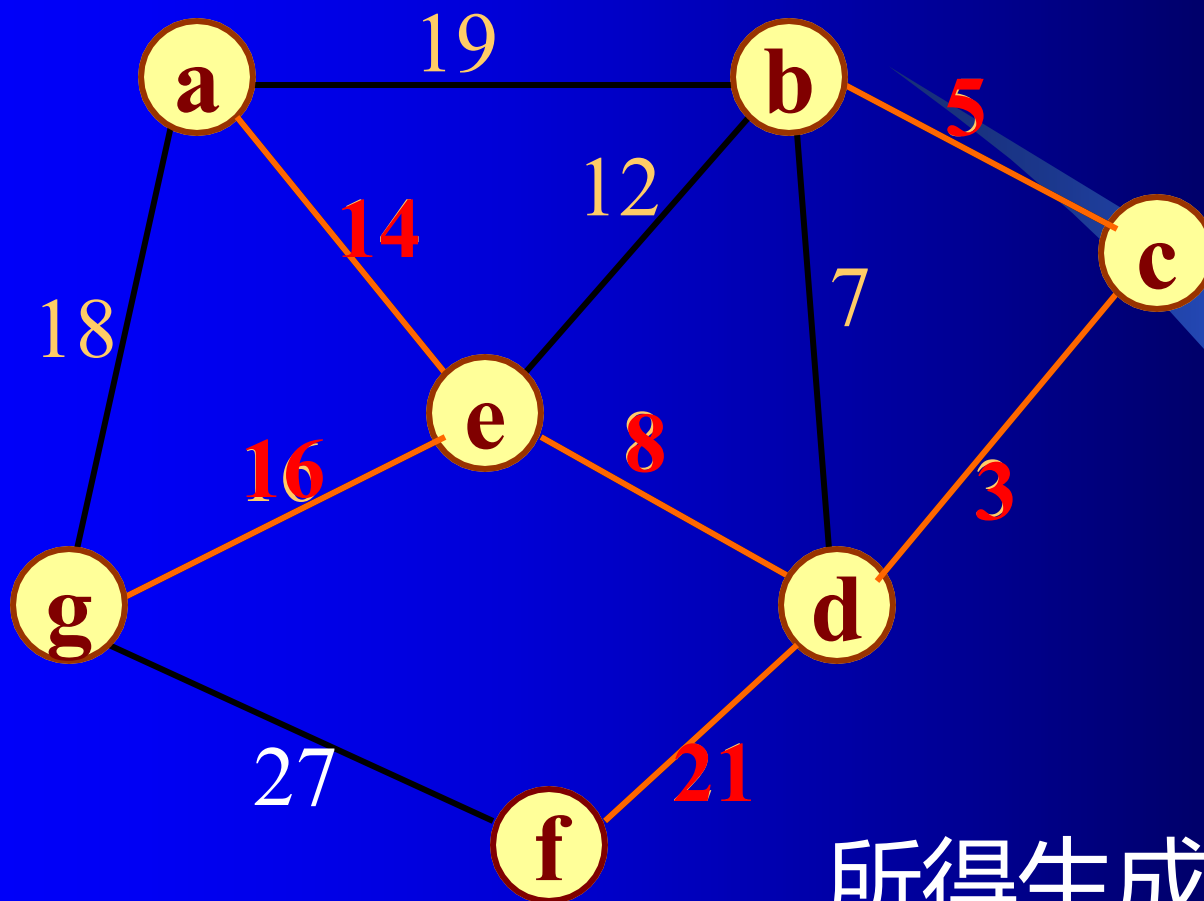
MST性质：

- 假设 $N=\{V,\{E\}\}$ 是一个连通网， U 是顶点集 V 的一个非空子集。若 (u,v) 是一条具有最小权值的边，其中 $u\in U$, $v\in V-U$,则必定存在一棵包含边 (u,v) 的最小生成树。
- 证明（略）。

普里姆算法的基本思想:

取图中任意一个顶点 v 作为生成树的根，之后往生成树上添加新的顶点 w 。在添加的顶点 w 和已经在生成树上的顶点 v 之间必定存在一条边，并且该边的权值在所有连通顶点 v 和 w 之间的边中取值最小。之后继续往生成树上添加顶点，直至生成树上含有 $n-1$ 个顶点为止。

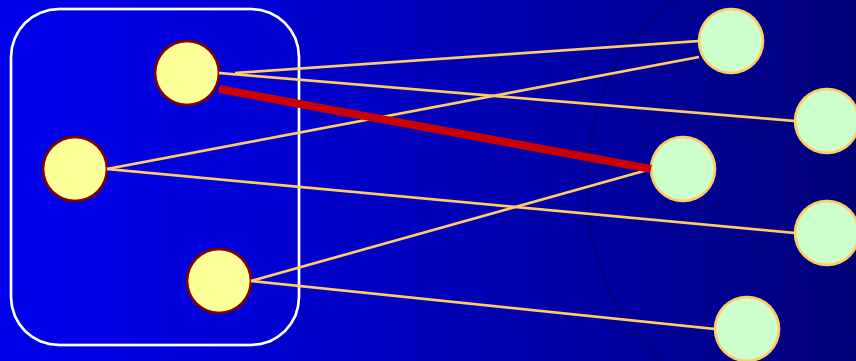
例如:



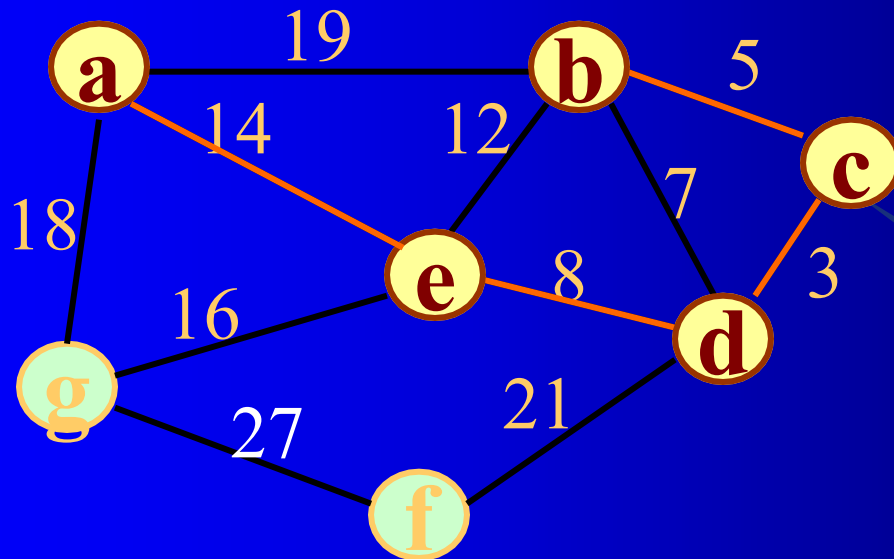
所得生成树权值和

$$= 14 + 8 + 3 + 5 + 16 + 21 = 67_{35}$$

一般情况下所添加的顶点应满足下列条件: 在生成树的构造过程中, 图中 n 个顶点分属两个集合: 已落在生成树上的顶点集 U 和尚未落在生成树上的顶点集 $V-U$, 则应在所有连通 U 中顶点和 $V-U$ 中顶点的边中选取权值最小的边。



例如:



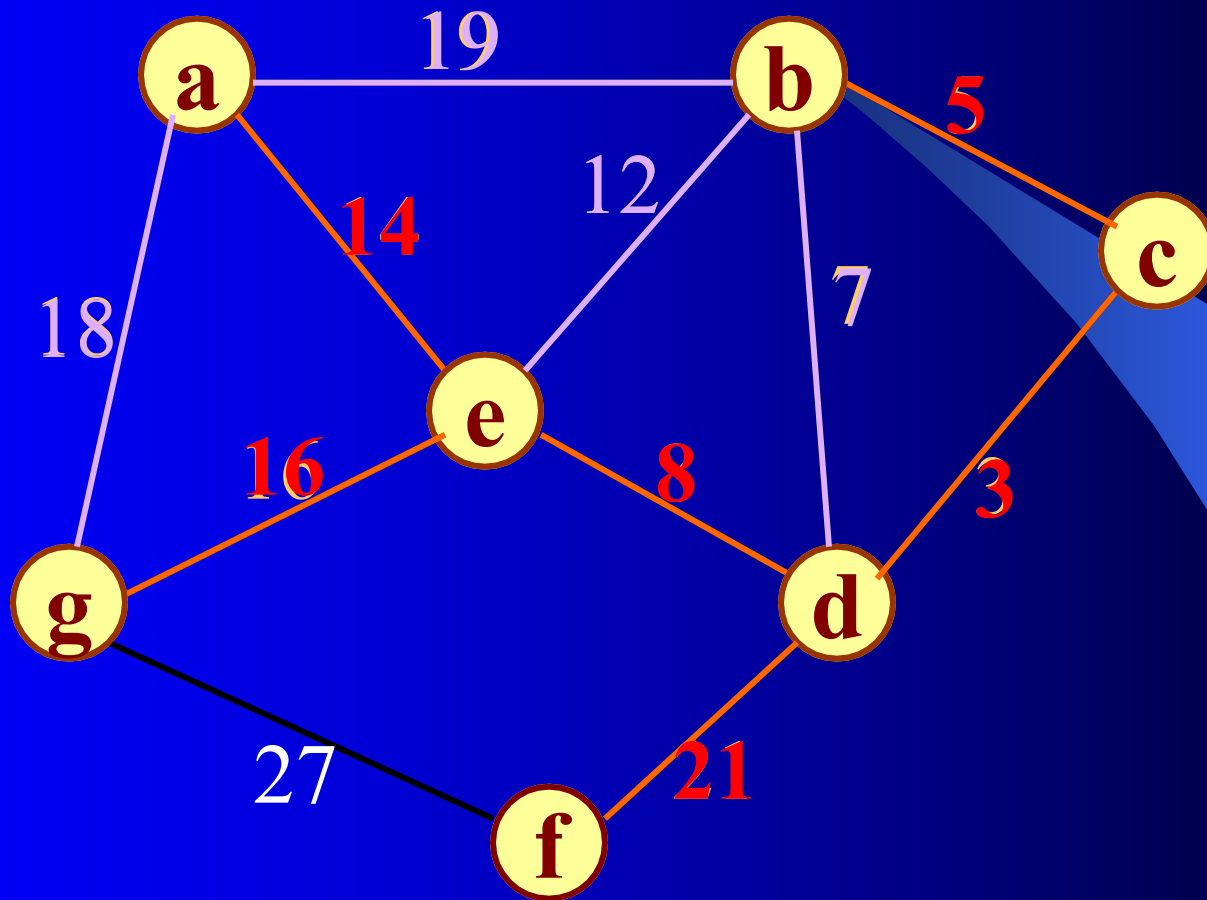
	0	1	2	3	4	5	6
closed							
edge							
Adjvex		c	d	e		d	e
Lowcost		5	3	8	14	21	16

克鲁斯卡尔算法的基本思想：

考虑问题的出发点：为使生成树上边的权值之和达到最小，则应使生成树中每一条边的权值尽可能地小。

具体做法：先构造一个只含 n 个顶点的子图 SG ，然后从权值最小的边开始，若它的添加不使 SG 中产生回路，则在 SG 上加上这条边，如此重复，直至加上 $n-1$ 条边为止。

例如:



比较两种算法

算法名	普里姆算法	克鲁斯卡尔算法
-----	-------	---------

时间复杂度	$O(n^2)$	$O(e \log e)$
-------	----------	---------------

适应范围	稠密图	稀疏图
------	-----	-----

请务必写出自己的模版！

再次提醒:

调课三周

(11/6, 11/13, 11/20)

附：贪心算法练习题：

- 1045 Fire Net
- 1050 Moving Tables
- 1051 Wooden Sticks
- 1052 Tian Ji -- The Horse Racing
- 1053 Entropy
- 1054 Strategic Game
- 2037 今年暑假不AC
- 1076、1203、1204、1239、1579、1730、2285
- 最小生成树：1102、1301、1162、1233

ACM,
天天见!

