

2010 年湖南师范大学 12 月份 月赛解题报告

By Huzijin

题目总纲

简单题：

ProblemA , Problem B , Problem C

中等题：

ProblemD , ProblemE , Problem F

难题（其中 I 由中南林罗四伟提供）：

ProblemG , ProblemH , ProblemI

ProblemA

- 转载自 : Coreforce Contest45 ProblemA
- 关键字 : C 语言 , 字符串 , 取模
- `char Month[][30]={"January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December" } ;`
- 当前月份为 x , 再过 k 月后就是 `Month[(x+k)%12]`, 打印即可。

ProblemB

- 转载自：2010 亚洲区域赛杭州赛区真题
- The Number Of Different Bit between A and B :
- $F(A,B) = \{k \mid k \text{ 表示 } A \& B \text{ 中 } 1 \text{ 的个数} \}$
- 也可以把 A 与 B 都装换成二进制，去比较
- `for(i = 0 ; i < 32 ; ++ i)`
- `if(BIT(A,i)==BIT(B,i)) ++ answer ;`
- `#define BIT(x,i) = ((x)>>(1<=(i)<=31)&1)`

ProblemC (原创)

- 给定 N , 求

$$S(n) = \sum_{i=1}^n \lfloor \sqrt{i} \rfloor$$

- 若 $f(x)=i$, 则 x 可取 $\{i^2, i^2+1, \dots, (i+1)^2-1\}$.
- 一共有 $(i+1)^2 - i^2 = 2*i+1$ 个 x , 使得 $f(x) = i$.

- 设 $a = \max(\{k \mid k*k \leq n\})$, 则有计算式:
- $$(n - a * a + 1) * a + \sum_{i=1}^{a-1} i * (2 * i + 1)$$

ProblemD

- 原创题：基础算法
- 关键字：BFS 或者 直接模拟
- 假设当前时间是 T ，Jerry 到达了 (tx, ty) ，用 BFS 求出 Tom 当前可以到达的所有格子，设为集合 A 。如果 (tx, ty) 在 A 中，则被抓到了，否则考虑下一秒
- 对 Jerry 很好处理，关键处理 Tom。

ProblemD

- 可以参考一下源代码：

```
■ int  dx[] = {1,0,-1,0} , dy[] = {0,1,0,-1} ;
■ for(int step = 0 ; step < K ; ++ step){
■     jx += dx[route[step]] ; jy += dy[route[step]] ;
■     memset(vis1 , 0 , sizeof(vis1));
■     for(int i = 0 ; i < N ; ++ i)
■         for(int j = 0 ; j < M ; ++ j)
■             if(vis0[i][j]){
■                 for(int k = 0 ; k < 4 ; ++ k){
■                     int nx = i + dx[k] ;
■                     int ny = j + dy[k] ;
■                     if(nx<0||ny<0||nx>=N||ny>=M||bd[nx][ny] == '*') continue ;
■                     vis1[nx][ny] = 1 ;
■                     if(vis1[jx][jy]) return true ;
■                 }
■             }
■     for(int i = 0 ; i < N ; ++ i)
■         for(int j = 0 ; j < M ; ++ j)
■             vis0[i][j] = vis1[i][j] ;
■ }
```

ProblemE

- 原创题：高级数据结构
- 关键字：线段树，树状数组，统计。
- 对三元组 (i, j, k) ，每次确定 j ，分别统计出 $\text{front}[j]$ 和 $\text{back}[j]$ 。
- $\text{front}[j]$ 表示 $S[1..j-1]$ 中小于 $S[j]$ 的个数。
- $\text{back}[j]$ 表示 $S[j+1..n]$ 中大于 $S[j]$ 的个数。
- $\text{For}(\text{answer} = 0, i = 1; i \leq n; ++i)$
- $\text{answer} += \text{front}[i] * \text{back}[i]$

ProblemE

- 统计 `front[j]` 和 `back[j]` 的过程和求逆序对的过程类似。（线段树和树状数组）
- 对低年级的小朋友们，建议先学好《数据结构》，然后再学高级数据结构，可以查阅历届国家集训队论文：线段树有关章节
- 学习流程：C 语言 -> 数据结构 -> 基础算法 -> 高级数据结构，也可跳过基础算法。

ProblemF

- 原创题：算法，图论
- 关键字：最短路，Floyed 算法，矩阵乘法，动态规划（可能算难题）。
- 首先回顾 Floyed 算法：
- $f[i,j]$ 表示从 S 到 j 经过 i 条边的最短路。
- $f[i,j] = \min(f[i-1, k] + G[k,j]) \quad 1 \leq k \leq n$
- 将 $f[i,1], f[i,2], \dots, f[i,n]$ 记做一个向量 $F(i)$ 。

Problem F

- 可以看出上述递推式是一个这样的关系：
- $F(i) = G * F(i-1)$
- 这里我们把递推式中的 + 看成矩阵乘法中的 + , min 看做矩阵乘法中的求和操作。
- 定义矩阵 " 乘法 " ：

$$(A * B)_{ij} = \underset{k=1}{\overset{n}{Min}} (A_{ik} + B_{kj})$$

Problem F

- 可以证明定义的“乘法”满足结合律
- 即： $(A * B) * C = A * (B * C)$
- 这样递归式最终写成了：
- $F(i) = F(i-1) * G$ ；
- 递推有： $F(i) = F(i-1) * G = \dots = F(0) * G^n$
- $n \leq 100000$, $row \leq 100$, $col \leq 100$
- 必须用快速幂计算。
- 这样求出来的 $F(i)$ 是正好经过 i 条边的最短路

ProblemF

- 而题目是求一条最短路，边数在 $[a,b]$ 之间。
- 必须进行如下装换：
- $$F(i) = F(0) * G^a * G_0^{b-a}$$
- 这里 G 就是邻接矩阵
- G_0 是邻接矩阵中赋值 $G[i,i]=0$ ，后的 G 。
- 其实 $F(0)*G^n$ 是求严格 n 条边最短路，
- 而 $F(0)*G_0^n$ 是求边数不超过 n 的最短路。

ProblemF

- 这里提供一些参考资料：
- 2008 国家集训队论文：《矩阵乘法在信息学中的应用》
- 本题命题思路来源于上述论文。
- 可以在网上查询一下几个关键字：
- 快速幂，最短路算法。
- 对低年级小朋友可能有点难度，等大家学习了算法后，可能会好一点。

ProblemG

- 转载题：国集论文《Hash 在信息学中的应用》杨弋
- 关键字：二分搜索，字符串 Hash.
- 具体细节参考论文。

ProblemH

- 原创题：图论。
- 关键字：二分，网络流，最小割，0/1 分数规划，dinic 算法。
- 知识要求比较多。学习网络流之后，能够弄懂这些内容。
- 命题思路来源：《最小割模型在信息学中的应用》 胡伯涛。

ProblemH

- 求一个子图 $G' = (V', E')$
- 使得 P/C 最大

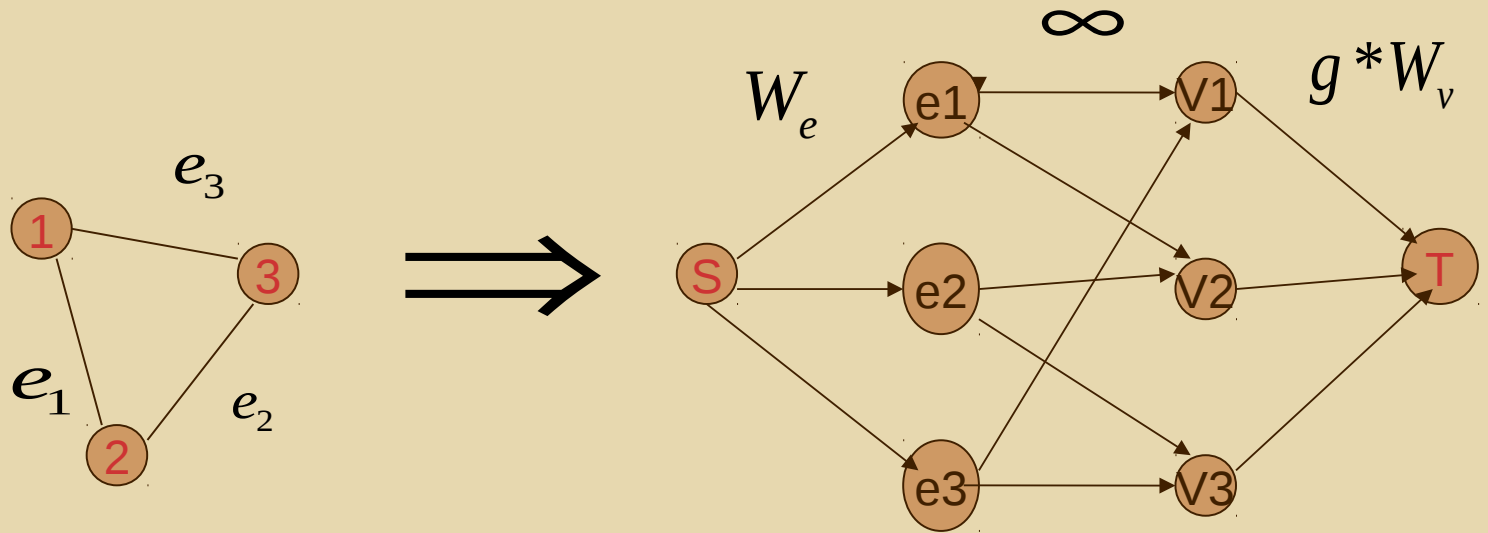
$$g = \frac{P}{C} = \frac{\sum_{e \in E'} W_e}{\sum_{v \in V'} W_v}$$

- 二分搜索参数

$$f(x) = \text{Max}(\sum_{e \in E'} W_e - g * \sum_{v \in V'} W_v)$$

ProblemH

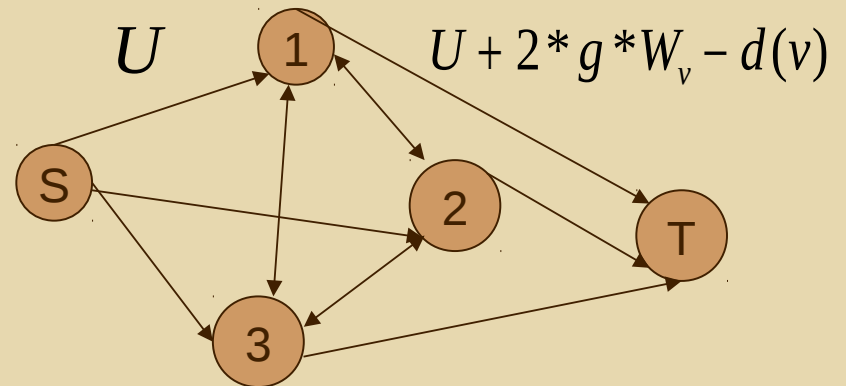
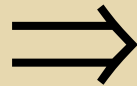
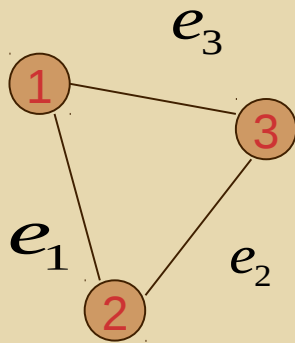
- 提供两种建图思路：
- 思路一：最大权闭合图



$$f(x) = \sum_{e \in E} W_e - C[S, T]$$

ProblemH

■ 2. 直接构图



$$d(v) = \sum_{v \text{ 是 } e \text{ 的邻接点}} W_e$$

$$f(x) = \frac{U * n - C[S, T]}{2}$$

ProblemH

- 具体资料参考国集论文：
- 《最小割模型在信息学中的应用》

Problem I

- 首先应注意到的一点是：不管怎么交换， $X+Y$ 的值是不变的！
- 考虑 $X*Y$ 在什么情况下能够尽可能的大呢？显然，两数越接近越好，并且当 $X=Y$ 时 $X*Y$ 是最大的，自行证明。

Problem I

- 于是很自然的贪心策略：1. 贪心使得经过 S 次交换 $X \geq Y$ 且 $|X - Y|$ 尽可能小，求得此时 $X * Y$ 的值 $res1$ 。2. 贪心使得经过 S 次交换 $Y \geq X$ 且 $|X - Y|$ 尽可能小，求得此时 $X * Y$ 的值 $res2$ 。
- 最终答案即为 $\max(res1, res2)$ 。
- 注意：需要大数乘法运算。