



唐山一中      任一恒

# 完美算法



**压缩**



**节省空间**

搜索



一千年以后 刘汝佳



**更快速方便**

## 2007 年部分应用非完美算法效果不错的题目

比赛	题目	标准算法	可采用算法
NOI	追捕盗贼	树搜索分治	贪心
CTSC	激光坦克	/	智能贪心
	矩阵	网络流构造	贪心随机
IOI	船帆	/	贪心调整
冬令营	剪刀石头布	网络流	贪心调整

一、随机算法

二、贪心算法

三、抽样测试法

四、模拟退火算法

五、等等算法 ^\_^

### 三、抽样测试法

抽样，即从统计总体中，任意抽出一部分单位作为样本，并以其结果推算总体的相应指标。在某些问题中，需要让我们检查一系列测试元  $s$ ，如果  $s$  中的某个测试元满足了某个条件，那么则说  $s$  满足了某个性质。在大度数情况下，我们需要将  $s$  中的测试元一个一个的进行验证，才能确定  $s$  是否满足该性质。但是如果  $s$  满足如下性质，要不  $s$  中不含满足条件的测试元，要不满足某条件的测试元很多，则可以直接选取几个具有代表性的测试元进行测试，通过这几个测试元来确定  $s$  是否满足该性质。

# 质数检验

对于一个整数  $n$ ，设  $n-1=d*2^s$  ( $d$  是奇数)，对于给定的基底  $a$ ，如果存在  $a^d \equiv 1 \pmod{n}$  或者对于  $0 \leq r < s$ ，存在  $a^{(d*2^r)} \equiv -1 \pmod{n}$  则称  $n$  是以  $a$  为基底的强伪质数。

质数  $n$ ，基底  $1 \cdots n-1$ ，必为强伪质数

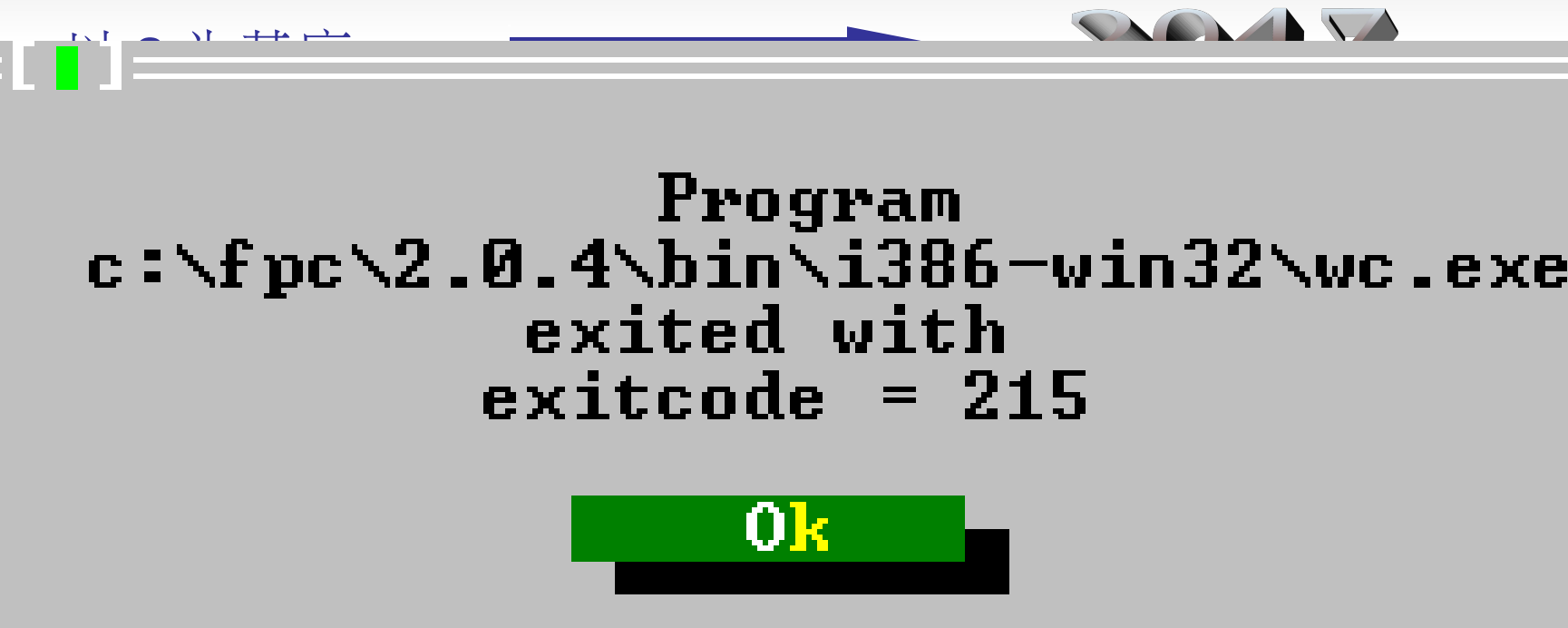
合数  $n$ ，基底  $1 \cdots n-1$ ， $< 1/4$  为强伪质数

# 质数检验

随机抽取——效率不高！

固定取样——最小质数。

质数检验



Program  
c:\fpc\2.0.4\bin\i386-win32\wc.exe  
exited with  
exitcode = 215

Ok

## 例题 紧急修复（百度之星 2007）

某市的  $k$  家公司的计算机系统全部瘫痪，要在  $T$  小时之后才能自动修复，每家公司每小时都在受到损失，第  $i$  家公司每小时受损为  $P(i)$ ，现在派遣  $n$  只维修队进行抢修，力求在自动修复之前将损失降到最小。

城市被划分为  $r \times c$  的网格，现给出了第  $i$  个公司的坐标  $(r(i), c(i))$ ，该公司的受损程度  $B(i)$  队·小时。还给出了每个维修队的初始坐标，每小时每个维修队可以移动（最多  $s$  格），也可以维修它所处所在的格子中的公司，现在希望你设计一种方案使损失降低到最小。



# 分析

**全局分析：每步产生一个全局最优方案**

**估价函数要求高，实现难度大**

**舍弃**

**极端想法：让所有队伍依次修理每个公司**

# 实现

- 1、预处理：计算每家公司到每个点的距离。
- 2、安排修理的顺序。
- 3、按照修理顺序计算每个修理队的活动（能参与维修的赶过去）。
- 4、计算损失。

# 修理顺序的选择

关键→修理顺序。

每种顺序损失的计算复杂度低

模拟退火。

# 模拟退火

模拟退火算法来源于固体退火原理，将固体加温至充分高，再让其徐徐冷却，加温时，固体内部粒子随温升变为无序状，内能增大，而徐徐冷却时粒子渐趋有序，在每个温度都达到平衡态，最后在常温时达到基态，内能减为最小。根据 **Metropolis** 准则，粒子在温度  $T$  时趋于平衡的概率为  $E^{-\Delta E/(kT)}$ ，其中  $E$  为温度  $T$  时的内能， $\Delta E$  为其改变量， $k$  为 **Boltzmann** 常数。

# 模拟退火

用固体退火模拟组合优化问题，将内能  $E$  模拟为目标函数值  $f$ ，温度  $T$  演化成控制参数  $t$ ，即得到解组合优化问题的模拟退火算法：由初始解  $i$  和控制参数初值  $t$  开始，对当前解重复“产生新解→计算目标函数差→接受或舍弃”的迭代，并逐步衰减  $t$  值，算法终止时的当前解即为所得近似最优解。退火过程由冷却进度表 (Cooling Schedule) 控制，包括控制参数的初值  $t$  及其衰减因子  $\Delta t$ 、每个  $t$  值时的迭代次数  $L$  和停止条件  $S$ 。

# 模拟退火基本思想

- (1) 初始化：初始温度  $T$  (充分大)，初始解状态  $S$  (是算法迭代的起点)，每个  $T$  值的迭代次数  $L$
- (2) 对  $k=1, \dots, L$  做第 3 至第 6 步：
- (3) 产生新解  $S'$
- (4) 计算增量  $\Delta t' = C(S') - C(S)$ ，其中  $C(S)$  为评价函数
- (5) 若  $\Delta t' < 0$  则接受  $S'$  作为新的当前解，否则以概率  $\exp(-\Delta t'/T)$  接受  $S'$  作为新的当前解。
- (6) 如果满足终止条件则输出当前解作为最优解，结束程序
- (7)  $T$  逐渐减少，且  $T \rightarrow 0$ ，然后转第 2 步。

# 模拟退火注意点

(1) 温度  $T$  的初始值设置问题

关联损失

(2) 退火速度问题

$L=20$

(3) 温度管理问题

$\text{Div } 2$

# 小技巧

## 优化初始解

$$B(I)/P(I) < B(I+1)/P(I+1)$$

生成新解：翻转两边 / 翻转中间

随机 P Q

$P > Q$  ( $w_1, w_2, \dots, w_q, w_{q+1}, \dots, w_p, \dots, w_n$ )  
( $w_q, w_{q-1}, \dots, w_1, w_{q+1}, \dots, w_n, \dots, w_p$ )

$P > Q$  ( $w_1, w_2, \dots, w_p, w_{p+1}, \dots, w_q, \dots, w_n$ )  
( $w_1, w_2, \dots, w_q, w_{q-1}, \dots, w_p, \dots, w_n$ )



# 效果

最佳	本程序	最佳	本程序	最佳	本程序
6060	6060	31366	32268	48914437	53810290
2680	2685	59367	60881	72336795	81953952
1530	1542	43011361	43011361	67911978	17180788
2920	3000	17841301	17841301	63405829	174311659
67	67	43970509	43970509	63637800	620868800
8511	9042	64210356	64210356	52417600	129294400
9988	11034	14985	18753	90282993	104940360
6541	7176	51523	59556	2904375	4303806
12484	13761	66143	72772	6022918	6781423
11026	10977	381800	389800	2152200	2174600

# 总结

应用广泛。

时空消耗低。

编程复杂度低。

思维容易理解。

