IOI2014 中国国家集训队作业 1 解题报告

慈溪中学 周以凡

2013 年 10 月 6 日

Contest

1 A Linking Loader (ACM/ICPC World Finals 2003 G)

1.1 题目大意

给出 3 种语句 D、E、C。

其中 D 表示定义一个符号的值，E 表示给一个符号编号，C 表示给一些数值分配地址。

其中 C 语句中的$符号表示将后一个数字所代表编号的符号的两位放入$符号指示的内存地址以及后面一个。

最后统计符号的值以及用校验和计算内存的值。

D 语句数量不超过 100。

1.2 关键字

模拟

1.3 骗分 1（25%）

不考虑 C 语句，仅仅只判断 D 和 E 语句，内存中不存在值，校验和为 0。正确维护符号名称和值即可。

用平衡树等数据结构存储可以做到更优秀的时间复杂度，但不影响得分。

【时间复杂度】 O(nS+M) O(nlogS+M) 【空间复杂度】 O(n+S+M)[1]

1.4 骗分 2（另外 10%）

仅考虑 C 语句，忽略所有 C 语句中的符号引用，此时将所有符号的值视为 0。

【时间复杂度】 O(n+M) 【空间复杂度】 O(n+M)

1.5 标准算法（100%）

在前面两个算法合并的前提下，考虑 D 语句存在向后引用的情况。用数组暂存内存的值，等所有语句处理完之后，对于每个符号的所有引用，修改内存中的某些值，最后再计算校验和即可得到全部分数。

题目中说明了“D”语句的数量不超过 100，但没有规定“E”语句的数量，从理解角度看应该少于 256，因此数据中每个模块中的该语句数量少于 256。对于此题 C++选手应该会大量使用 STL 库，也是对模板使用能力的考察。

【时间复杂度】 O(nS+M) O(nlogS+M) 【空间复杂度】 O(n+S+M)

[1] 此处及 1.4, 1.5 中，n 表示操作个数，S 表示不同的符号个数，M 表示内存单元的大小。

2 Glenbow Museum (ACM/ICPC World Finals 2008 F)

2.1 题目大意

用 R(90°)和 O(270°)表示一个直角多边形内角的度数，判断对于给定长度，有多少用 R 和 O 组成的序列能画出闭合的图形，并且在图形内部有一点能看到所有边界。

2.2 关键字

数学，排列组合，DP

2.3 特判（0%）

注意到 n 的个数小于 4 或者 n 为奇数时无解，可以直接输出 0，但很可惜本题由于是多组测试数据，此方法并不能得到任何分数，但以下的算法使用此特判可以做出一定优化或者避免答案错误。

【时间复杂度】 O(1) 【空间复杂度】 O(1)

2.3 暴力 1（10%）

本题至少想到，应该将**在图形内部有一点能看到所有边界**这个条件转化为不存在连续的 O，那么可以使用搜索来确定序列的个数，按照角度和判断时候闭合即可。需要一定优化才能拿到第二个点的 5%的分数。

【时间复杂度】 O(2^n) 【空间复杂度】 O(n)

2.4 暴力 2（40%）

考虑到凸多边形的外角和为 360°，我们将 270°视为-90°，所有的角度和为 360°，那么 R 的个数应该比 O 多 4 个，我们枚举这多出的 4 个 R 放的位置，然后统计答案。程序做出一定常数优化可以获得 55%的分数。

【时间复杂度】 O(n^4) 【空间复杂度】 O(n)

2.5 标准算法 1 DP （100%）

现在知道 R 的个数比 O 多 4 个，并且不能存在连续的 O，我们使用 DP 来求解这道题。令 f[i][j][k]表示，前 i 个位置填放完毕，放置了 j 个 R，最后一个是否为 O 的方案个数。考虑开头是否为 O 的情况，最后统计答案即可。

【时间复杂度】 O(n^2) 【空间复杂度】 O(n^2)

2.6 标准算法 2 排列组合（100%）

回忆数学中的排列组合，我们发现其实可以直接算出答案，分别考虑开头是否为 O 的情况，设 R 的个数为 K，答案为：

C(K, 4) + C(K - 1, 4)

【时间复杂度】O(1) 【空间复杂度】O(1)

3 Pollution Solution (ACM/ICPC World Finals 2013 J)

3.1 题目大意

给出一个在 x 轴上方，点数不超过 100 的多边形以及一个半径不超过 1000 的半圆，求多边形被半圆覆盖的面积。

3.2 关键字

计算几何，多边形面积，自适应 Simpson 积分

3.3 算法 1（约 30%）

数据中存在约 30%的 n=3 的情况，这时多边形为三角形，考虑三角形与圆的位置关系求出面积即可。

【时间复杂度】 O(n) 【空间复杂度】O(n)

3.4 标准算法 1（100%）

对比多边形面积的求法，我们可以将问题简化为求一个三角形被半圆覆盖的面积，然后根据两个顶点表示的向量的叉积的值，将各个面积加减得到答案。

此时，三角形的一个顶点必在圆心。我们可以分 4 类情况讨论。

第一类，三角形其余两个顶点均在半圆内部，那么答案就是三角形的面积；

第二类，三角形一个顶点在半圆外部，那么答案化为一个三角形和一个扇形的面积和；

第三类，三角形两个顶点均在半圆外部，并且第三条边与圆弧交点小于两个，那么答案就是一个扇形的面积；

第四类，三角形两个顶点均在半圆外部，并且第三条边与圆弧有两个交点，那么答案就是一个三角形和两个扇形的面积和。

对于第三类和第四类的情况可以用点到线段的距离来判定，但是比较麻烦，其实可以将后三类情况一起判断，直接使用解二次方程的方法判断和圆弧交点的个数以及交点的位置。

【时间复杂度】 O(n) 【空间复杂度】O(n)

3.5 标准算法 2（100%）

对于本题这种求解面积的问题，我们可以采用**自适应 Simpson 积分**。

定义函数 $f(a)$，表示直线 $x = a$ 所截取的我们所要求的答案区域的长度。

定义函数 $Calc(l, r)$，表示 x 坐标在 (l, r) 区间内的答案（区间开闭不影响答案），

$Calc(l, r) = Calc(l, mid) + Calc(mid, r)$，

其中 mid 为 l 与 r 的平均值。

因为 l 与 r 均为实数，所以函数递归无法退出。我们首先对函数所求区域的面积进行估价。

我们将区间内的平均 $f(a)$ 函数值视为 $(f(l) + f(r) + 4 * f(mid))/ 6$ 。该式的推导超出本文讨论范围，此处不赘述。

那么区间 (l, r) 的期望面积 $S(l, r) = (f(l) + f(r) + 4 * f(mid))/ 6 * (r - l)$ 。

如果 $S(l, r)$ 与 $S(l, mid) + S(mid, r)$ 相差不超过我们所定的 eps（一般为 1e-9），函数直接退出，此时答案已经足够精确，所以不需要继续分割区间计算了。

因此，首先对坐标轴进行随机旋转，使得所有交点的 x 坐标两两不同。

然后离散所有交点的 x 坐标，对于相邻的 x_i, x_j $(i + 1 = j)$，调用函数 $Calc(x\_i, x\_j)$ 计算，将面积相加即可。

【时间复杂度】 $O(n\log EPS^{-1})$ 【空间复杂度】$O(n)$

感谢福建黄豪硕同学提供此方法。