

# ACM/ICPC World Finals 2013 E Harvard 解题报告

——武汉二中 黄斐

## 1. 题目简述

有一台计算机拥有  $b$  块不同的内存区域，每块能储存  $s$  个变量。有一段程序，操作有 2 种，循环指令和访问变量指令。如果访问的变量在 1 号内存区域中则需要一条指令，不在 1 号内存区域中则需要操作 BSR 表明区域位置（除非 BSR 已经是想要的值），然后再访问（两条指令）。给定程序、内存块数、内存大小，求出一个变量到内存位置的映射，使得程序需要运行的指令数最少。其中  $b$ 、 $s$ 、变量个数  $\leq 13$ ，程序长度不超过 1000，答案在 long long 范围内。时间限制 10s。

## 2. 解题思路

首先设变量个数为  $n$ ，程序长度为  $l$ 。

因为每次访问变量始终都要使用 1 条指令，所以这题实际上是要求切换 BSR 的次数，也就是相邻 2 次访问变量(忽略在 1 号内存块上的变量)不在同一内存块上的最小值。

变量个数非常少，考虑搜索搜出每一个变量的位置，然后代入更新答案。

复杂度为  $s^n * l$ ，显然裸做是会超时的，从两个方面考虑优化。

搜索剪枝：这道题不能搜一部分，算出可能的最小解，使用最优化剪枝，所以需要根据题目特点进行剪枝。剪枝 1，1 号内存如果能填满，则一定会填满（否则将任意在其他内存块的变量移到 1 号内存块，答案不会更差）。剪枝 2，使用率最小的 2 块内存区域和一定小于  $s$ （否则将这 2 块合为 1 块，答案不会更差）。这样剪枝以后最坏情况下需要更新答案的次数将近 300W 次。

减少每次验算的复杂度：如果每次都要带入程序重算的话，每次验算的复杂度为  $O(l)$ ，不能接受。观察可以发现答案只和每 2 个变量相邻的次数有关系，也就是我们可以开出一个数组  $t[i][j]$  记录  $i$  号变量和  $j$  号变量相邻的次数，这样每次搜完后都可以再  $O(n^2)$  的复杂度内更新答案。但是注意到一点，计算相邻次数的时候要忽略在 1 号内存块上的变量，也就是说我们必须要在把第 1 块内存块上的变量搜完才能计算这个数组。预处理的复杂度为  $O(l * n^2)$ ，也可以优化到  $O(l)$ ，由于不是算法瓶颈所以没有优化。

重新整理思路，就是说首先搜谁在 1 号内存块，预处理  $t$  数组，然后搜剩下的元素，然后更新答案。这样的复杂度大约是  $C(n,s) * l * n^2 + 300W * n^2 \leq 10E$ ，足够能通过此题。

## 3. 感想

这道题在比赛的时候做出来的队很少，但是其实不是这题很难，而是复杂度难以估计，只有写后试试才能知道计算次数，不能保证是否可以通过。