
思路题解 丰收祭前的游戏

sp60 liul_sp60@163.com

将 `example.csv` 文件放在 `exe` 文件同一级目录，点击 `exe` 执行。如遇见不能运行问题，望发邮件告知。 邮箱：liul_sp60@163.com

我生成的`.exe`已经拷贝一份在压缩包第一级目录里面，双击运行就可以了。

情景建模：

转化为图，//人数==节点，好友关系==存在连通的边。求图中长度为4,6,8,10,12,14的简单环(不重复经过同一个节点的环)的个数。再去重（不同的环可能有着相同的节点）。

代码时间与空间

系统平台：Windows10_64 位、i5-4210U 8GB 内存、visual studio 2019。

设置 debug、release 均设置为速度优先，O2 速度优先，全局优化。

.csv 数据文件中 A、B 部落的人数分别为 256、640。超出 14 不搜索，release 版本消耗时间 10s 左右。

.csv 数据文件中 A、B 部落的人数分别为 1344、1344。超出 14 不搜索，release 版本消耗时间 280-300s 左右，在性能探查时峰值内存约为 145MB。

将 `example.csv` 文件放在 `exe` 文件同一级目录，点击 `exe` 执行。

部落 A 人数：numTA

部落 B 人数：numTB

总人数： numTotalAB

部落内部无直接连接，仅在部落间存在直接连接。（tanner 图，没找到规律去重复的环，相同节点但节点顺序不同的环）

深度优先搜索所有的环。

Hash 去重（unordered_set）

方法: DFS+去重+剪枝。

去重策略一

以一个四个节点的环如 A1 —— B1 —— A3 —— B3 回到 A1 为例。有如下几种排列 但只算一个环。以 B 部落中的人为搜索起点的环和以 A 部落中的人为搜索起点的环一样多，且属于一个环，仅搜索 A 为搜索起点的环。降低了一半的搜索量。以 A 部落为起点直接用 DFS 搜索可搜索得到四种。

去重策略二

以 A1 为起点搜索，每搜索到一个 n 节点环，n 节点个数加 1（正反共搜索

两次，忽略第二个节点大于最后一个节点的情况，即忽略反向）（DFS 直至搜索到设定最大长度 `step` 的环 即把所有含有 A1 的环都给计数了，在此后的搜索中 A1 不参加搜索）

再以 A2 为起点搜索，此时遇见 A1 节点就跳过。重复上一步即每搜索到一个 `n` 节点环，`n` 节点个数加 1。

直至把 A 部落搜完。此策略保证越往后搜索规模越小搜索速度越快。

去重策略三

上述两个方法保证搜到不重复的环，不同的环采用快排序，把节点从小到大排序，连接成字符串。用 `unordered_set` 做映射去重。不同长度的环各自所在一个 `set`，程序中采用的是每搜索到一个环，判断该环的节点是否是已经搜索过的，是则不统计个数，否则该长度的名字的个数加一，并把该环加入相应的 `set`。

剪枝策略

通过对邻接矩阵能计算图中两个节点在几步内能互相访问到。AdjS2, AdjS3, AdjS4, AdjS5。比如初始节点为 A1 已经访问了十二个不同节点 当前节点访问至 A6。直接查询图中 A6 能否在两步内到达 A1，不能就减去这一枝，能就继续访问下去。 以此作为剪枝策略。

加速技巧

邻接矩阵是稀疏的，稀疏程度很高，采用邻接表存储邻接关系既节约存储空间又能加快搜索下一个节点。用数组存储会比链表快。

计算 两步三步四步五步的可访问点关系时采用稀疏矩阵乘法的条件搜索判断就行，不需要算乘法。只存储 0 和 1。

在当前给定的数据大小和关系中，两步三步四步五步的可访问点关系都是相对稀疏的。六步以上是相对稠密的。剪枝策略做到六步就可以了。计算过多的剪枝可能会影响效率。

用 `vector<type>` 存储数据会虽然比较方便，但不如用数组访问快。尽量换成数组存储。////二维数组采用指针数组存储。

解法写在一个类中，递归调用放在 `public` 中，仅在递归中访问的且需要记录的量、变量放进 `private`，减少递归函数中传递的变量。或者某些量采用全局变量。

本题中的 环的节点信息占用一个 `int` 直接存入 `set` 会浪费空间，可以采用两个节点合并成一个 `int` 型数据。`(temp [0] << 12)+ temp [1]` 这样会节约存储空间。同时不会改变存储的节点信息。

取舍权衡

标记已访问数组，几步内可访问点的关系时，用 `bool` 变量，位存储会节约内存，但直接用 `int` 存储 0 和 1，读取速度会更快。

考虑当前图的特殊关系。图连接关系不需要 `numTotalAB*numTotalAB` 大小， AdjS2 和 AdjS4 仅设置为 `numTA* numTA`。 AdjS3 和 AdjS5 设置为 `numTA* numTB`。甚至 AdjS2 和 AdjS4 可以再减小一半但会增加访问判断。

计划下一步的优化方向:

将 `example.csv` 文件放在 `exe` 文件同一级目录，点击 `exe` 执行。如遇见不能运行问题，望发邮件告知。 邮箱: liul_sp60@163.com