

学生实验报告



课程名称\_\_\_高级数据结构 \_

题目名称\_\_\_找出所有的社交集群\_\_

小组成员\_\_ 32001272-徐彬涵 \_\_\_

（学号-姓名）

\_ 32001269-徐若萱 \_

任课教师\_\_ \_

2021 年10月 17 日

# 零.完成情况

**PINTIA评测成绩：**



**成员分工及贡献情况：**

5分 徐彬涵：编写代码测试样例

5分 徐若萱：编写报告

# 一. 问题描述

# 题目：找出所有的社交集群

当你在社交网络平台注册时，一般总是被要求填写你的个人兴趣爱好，以便找到具有相同兴趣爱好的潜在的朋友。一个“社交集群”是指部分兴趣爱好相同的人的集合。你需要找出所有的社交集群。 输入格式： 输入在第一行给出一个正整数 N（≤1000），为社交网络平台注册的所有用户的人数。于是这些人从 1 到 N 编号。随后 N 行，每行按以下格式给出一个人的兴趣爱好列表：

K​i​​: h​i​​[1] h​i​​[2] ... h​i​​[K​i​​]

其中K​i​​(>0)是兴趣爱好的个数，h​i​​[j]是第j个兴趣爱好的编号，为区间 [1, 1000] 内的整数。 输出格式：

首先在一行中输出不同的社交集群的个数。随后第二行按非增序输出每个集群中的人数。数字间以一个空格分隔，行末不得有多余空格。 输入样例：

8 3: 2 7 10

1: 4

2: 5 3

1: 4

1: 3

1: 4

4: 6 8 1 5

1: 4

输出样例：

3

4 3 1

# 二.数据结构定义与分析

**分析问题**，根据分析写出数据结构定义

这是一道考察并查集的题目，可用建立父亲数组解决问题

# 三、算法分析

写出算法的主要思想

按照输入顺序对每个人进行编号，使用flag数组将第一次出现的爱好编号对应到第一次出现的爱好所属的人的编号，然后如果爱好不是第一次出现，则将当前用户编号与该爱好所属用户编号建立关系，从而一个一个建立用户关系，再利用并查集解法求出集群数量及每个集群的人数

# 四. 算法的时间复杂度分析

分析算法的时间复杂度以及分析过程

该算法将输入的每一个人根据兴趣进行并查集Merge操作，这段代码时间复杂度为O(n)，再进过三段对n的遍历来计算社群的数量及人数，这段复杂度也为O(n)，其中有个排序函数sort，时间复杂度为O(n\*log2(n)).

# 五、测试用例的选择原因以及测试结果

写出测试用例，详细阐述每个测试用例的选择原因，测试结果可以截图，（注意截图不要出现姓名、学号等信息）

一.测试样例

8

3: 2 7 10

1: 4

2: 5 3

1: 4

1: 3

1: 4

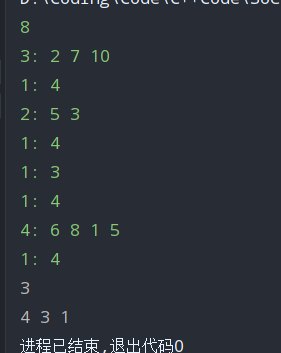
4: 6 8 1 5

1: 4

输出：

3

4 3 1



二．仅有一个社群的情况

5

3:1 2 3

2:3 4

2:4 5

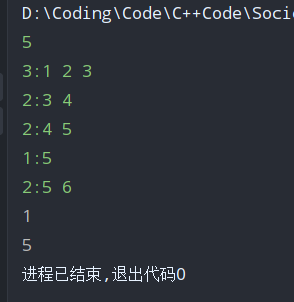
1:5

2:5 6

输出：

1

5



三．最大数量即1000个数组

<https://paste.ubuntu.com/p/tWSDYszT6T/>

给出生成样例脚本可自行测试

四．一人一个集群

6

1: 1

2: 2 3

3: 4 5 6

1: 7

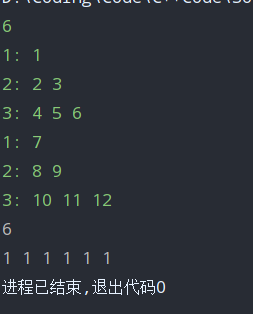
2: 8 9

3: 10 11 12

输出：

6

1 1 1 1 1 1



# 六、存在问题及可能原因

算法可能存在的问题，若有，请写出解决原因与方法。

暂未找到问题

# 七、附源代码以及代码相应注释

源代码

https://paste.ubuntu.com/p/jjp7Zv7mHd/

# 八、实验创新点或合理化建议

本次实验的创新之处以及是否可以有更好的建议使得题目更加有实用意义等。

可以进行再加上一个朋友的关系，让两个没有共同爱好的人也可以在一个社群，增加题目复杂