**代码结构**

问题简化为求图的无向环数量的问题，代码分为五大模块。

1. 读取CSV文件数据并进行处理，转化为unordered\_map，用来存储图的节点连接关系。
2. 对读取的数据进一步整理排序。
3. 剪枝，使用递归方法找到节点i三步之内可以连接到的所有节点。
4. 深度优先搜索并剪枝。如果节点数已经大于循环数减去剪枝数，那么根据步骤3生成的剪枝数组检查当前节点是否能在三步之内连接成环，不能则停止搜索。找到的环转化为字符串存于集合中，利用集合去重。
5. 根据存储环的集合的元素数量得到所要的结果，输出结果到屏幕和result.txt文件。

**代码测试环境**

在VS2015中，采用Release x64生成zte\_vs.exe。VS2015生产的文件在项目文件夹下x64/Release/zte\_vs.exe。

**使用方法**

1. 默认模式。将Example.csv放在zte\_vs.exe相同目录，使用命令行运行./zte\_vs.exe。默认数据文件是./Example.csv，循环次数上限是14。
2. 改变循环上限, ./zte\_vs.exe [循环次数上限]。如./zte\_vs.exe 10。
3. 改变循环上限, ./zte\_vs.exe [ [循环次数上限] 数据文件路径]。如./zte\_vs.exe 12 Example6.csv。
4. 结果可以通过控制台屏幕和./result.txt文件查看。

**测试结果**

使用i5-3210M 4G内存 win8 64位笔记本进行测试，配置比较低。

第一阶段测试完美通过耗时54.008s。

第二阶段12环在测试完美通过，耗时91.586s。

14环第二阶段由于存储换的数量巨大，导致内存不足，会在某一进度卡住，未能在windows下测试。

使用macbook pro 2.5GHz i5 16G内存 10.14.6Mojave系统进行测试，使用g++ -std=c++11 -O3 zte\_mac.cpp -o zte\_mac编译，./zte\_mac运行，测试14环第二阶段耗时1617.65s，时间略长，得分100分。

**其他**

时间比较匆忙，程序还有很多可以进一步优化的地方，文档也略显简陋，请老师见谅，如果有什么问题请联系我QQ 444183564。