密级: 公 开

信息工程大学

操作系统实验报告

Linux初识

单 位： 网络空间安全学院

队 别： 十一大队一队

学 号： 4042017036

姓 名： 虞 晨

学科专业： 网络工程

日 期： 2019 年 9 月 8 日

说 明

一、本实验的目的是帮助学员熟悉Linux下基本命令行使用、gcc编译器的使用等操作。

二、 要求学员使用C语言进行代码实现下面的题目。

**数据说明：**

总共包括3个数据文件：customer.txt, lineitem.txt和orders.txt。分别行数为100，1000和4000，列和列之间使用|来分隔。所有文件均为文本文件。

每一个顾客在customer表中有一个唯一的顾客编号c\_custkey，和对应的某个市场部门c\_mkgsegment。每个顾客可能会有任意多个历史订单，每个订单在orders表中有一行，它有一个唯一的订单号o\_orderkey和下单顾客编号o\_custkey和一个下单时间c\_orderdate。每个订单又由多件商品组成，每件商品在lineitem中有一行，记录了它所属的订单号l\_orderkey、价格l\_extendedprice及发货时间l\_shipdate。

**目标是找出每个订单延迟发货的商品的总销售额，以供销售方有针对性地分析优化。**如下图所示，我们将该问题抽象为以下的一个计算问题：每张表看作一个multi-set，对customer、orders和lineitem三张表建立笛卡尔集合，得到一个mnl行的一个集合（其中m,n,l分别是三张表的行数），然后对该集合按以下条件做过滤：

c\_mktsegment = ？ and c\_custkey=o\_custkey and o\_orderkey=l\_orderkey and o\_orderdate < ? and l\_shipdate > ? ：其中？是动态变参

对满足过滤条件的结果，按照l\_orderkey列的值进行分组，并对l\_orderkey相同的记录的l\_extendedprice做求和，最后结果按照l\_extendedprice求和值排序，并返回topn的结果，其中topn值也是动态变参。

该计算过程，也可以使用以下SQL来描述：

select

l\_orderkey,

o\_orderdate,

sum(l\_extendedprice) as revenue

from

customer,

orders,

lineitem

where

c\_mktsegment = ？

and l\_orderkey = o\_orderkey

and c\_custkey = o\_custkey

and o\_orderdate < ？

and l\_shipdate > ？

group by

l\_orderkey,

o\_orderdate

order by

revenue desc

LIMIT ？

**提交要求：**

需要提供5个文件（夹）

Readme.txt：编译和运行方式

Compile.sh：编译脚本

Run.sh: 执行脚本

Sourcecode 文件夹：包括所有source code和配置文件

可执行文件：可执行文件

**程序执行方式：**

选手需要提前将3个数据文件放到run.sh 相同目录，提交的程序将会以如下格式的命令运行、进行评测。

./run.sh customer.txt orders.txt lineitem.txt n n4个参数

第四个参数表示总共计算的次数

第5~8个参数为第一次计算时的4个参数

第9~12个参数为第二次计算时的4个参数

…

第1+4n~4+4\*n 个参数为第n次计算时的4个参数

四个参数分别对应以下的条件值：

c\_mktsegment = ？

o\_orderdate < ？

l\_shipdate > ？

LIMIT ？

1. S.

* 可以放一个docker.file

加分项：

* 多线程并行版本
* CUDA版本
* 不同并行度下的程序运行性能
* 其他优化手段

执行示例：

示例1：

./run.sh customer.txt orders.txt lineitem.txt 1 BUILDING 1995-03-29 1995-03-27 5

示例2：

./run.sh customer.txt orders.txt lineitem.txt 3 BUILDING 1995-03-29 1995-03-27 5 BUILDING 1995-02-29 1995-04-27 10 BUILDING 1995-03-28 1995-04-27 2

程序输出：所有结果按行输出，多列结果使用|分割

输出示例：

示例1：

l\_orderkey|o\_orderdate|revenue

249739810|1995-02-28|513890.13

180639074|1995-03-20|502044.66

202071367|1995-03-12|499760.62

31617348|1995-03-27|498949.38

461984355|1995-03-13|491959.42

示例2：

l\_orderkey|o\_orderdate|revenue

249739810|1995-02-28|513890.13

180639074|1995-03-20|502044.66

202071367|1995-03-12|499760.62

31617348|1995-03-27|498949.38

461984355|1995-03-13|491959.42

l\_orderkey|o\_orderdate|revenue

594317283|1995-02-11|457168.01

328093060|1995-02-03|453863.74

322499142|1995-02-25|427615.4

319693698|1995-02-05|425802.97

323689092|1995-02-23|416615.42

128919045|1995-02-28|414996.73

540656743|1995-02-23|411107.69

311469825|1995-02-21|402946.11

164796770|1995-02-27|401863.1

131043138|1995-02-20|399884.41

l\_orderkey|o\_orderdate|revenue

31617348|1995-03-27|498949.38

461984355|1995-03-13|491959.42

|  |  |  |
| --- | --- | --- |
| 一、基本情况 | | |
| 实验环境 | | 说明：使用的笔记本型号，虚拟机配置，Linux版本  使用的笔记本型号为：  Linux版本：  CentOS 6 64位 |
| 二、实验过程  （一）对目标功能在Windows系统下的实现  **1.1设计思路**  实验给出了三个表格，分别对应不同的信息，但又有相同的列可以级联，所以想到以下操作：  1、创造三个结构体：customer、orders、lineitem  2、对三个结构体进行级联生成新的结构体：select\_result,并对 extendedprice求和值排序。  3、根据要求输入limit、order date、ship date加以限制。  **1.2程序实现**  1.2.1 读入txt内容  利用fopen函数读入txt中的内容，针对txt中的无用项“|”，设定一个char型的c用于存取，对结构体内的值不产生影响。  1.2.2级联  在满足筛选条件如下时：  “cus[i].c\_custkey==ord[j].o\_custkey&&ord[j].o\_orderkey==item[k].l\_orderkey&&(strcmp(ord[j].o\_orderdate,order\_date)<0)&&(strcmp(item[k].l\_shipdate,ship\_date)>0)”  用select函数对result表进行赋值，根据要求进行冒泡排序并求和。  1.2.3输入与输出  在scanf函数下输入limit、order date、ship date，并将该三个参数的值在select函数中进行调用，进行输出时，当limit超出result表中有效组数时，输出会出现问题；加入一个判别条件（l.orderkey<0），超出部分输出null。  **1.3源码（见code文件）**  （二）对目标功能在CentOS下的实现  **2.1代码的改动**  Linux下传参有所变化：  对于每个main函数来说，int main(int argc,char\* argv[]);  argv是命令行参数个数 不带参运行，argc值为1；  argv是命令行参数；  **2.2脚本的编写：**  2.2.1 compile.sh  ***gcc -std=c99 -o experiment.out experiment.c***  编译experiment.c文件并生成对应的.out文件  2.2.2 run.sh  ***./experiment.out $@***  $@含义：所有参数列表。如"$@"用「"」括起来的情况、以"$1" "$2" … "$n" 的形式输出所有参数。  **2.3运行结果实例**      **2.4问题及解决**  问题1：在terminal里输入“./compile.sh”后报错“Permission denied”提示脚本权限不够。  解决办法：输入命令行：chmod 777 compile.sh  777代表，user,group ,others ,都有读写和可执行权限。  问题2：执行./run.sh时显示    解决方法：执行vi compile.sh | | |
|  | | |
| 三、总结部分  描述通过实验课学到的知识和建议。 | | |