**深 圳 大 学 实 验 报 告**

**课 程 名 称： 计算机系统(2)**

**实验项目名称： 实验环境配置与使用**

**学 院： 计算机与软件学院**

**专 业： 软件工程**

**指 导 教 师： 罗 胜**

**报告人： 张欣杰 学号： 2020151091 班级： 软工二班**

**实 验 时 间： 2022/3/17**

**实验报告提交时间： 2022/3/30**

**教务处制**

**一、 实验目标：**

熟悉Linux上C程序的编译和调试工具，包括以下内容：

1. 了解Linux操作系统及其常用命令

2. 掌握编译工具gcc的基本用法

3. 掌握使用gdb进行程序调试

**二、实验环境与工件**

**1.**个人电脑

2. Ubuntu操作系统

3. gcc

4. gdb

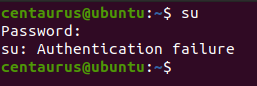
**三、实验内容与步骤**

1. 根据实验一：实验环境配置与使用.ppt熟悉Linux基本操作（P.1 – P.28）,然后根据以下过程创建用户：按照1.1~1.3完成并截图，截图需要有运行的命令及其结果。另外：**后面的题目必须在该新建用户下完成。（30分）**

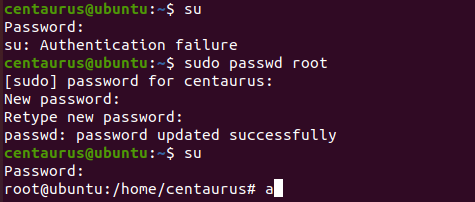
实验步骤：

* 1. 首先切换为超级用户

*$su*

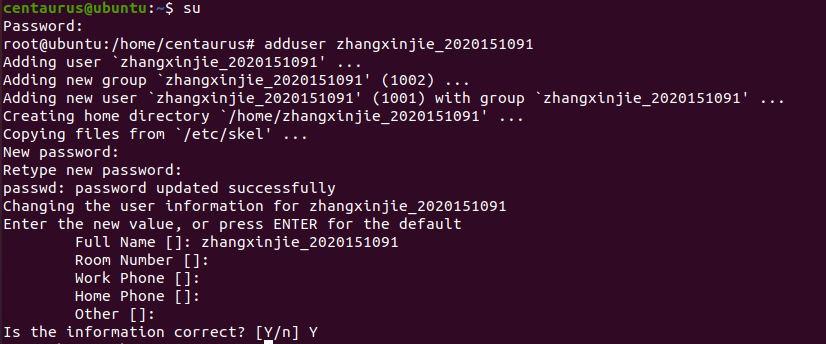


出现Authentication failure情况，使用`sudo passwd root`来给root重设密码：

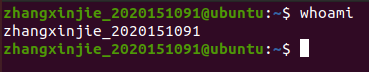


* 1. 参考以下命令创建新用户, 设置新建用户的密码，注意：只有设置了密码才能激活用户，否则无法以该用户身份登录

*#adduser zhangxinjie\_2020151091*



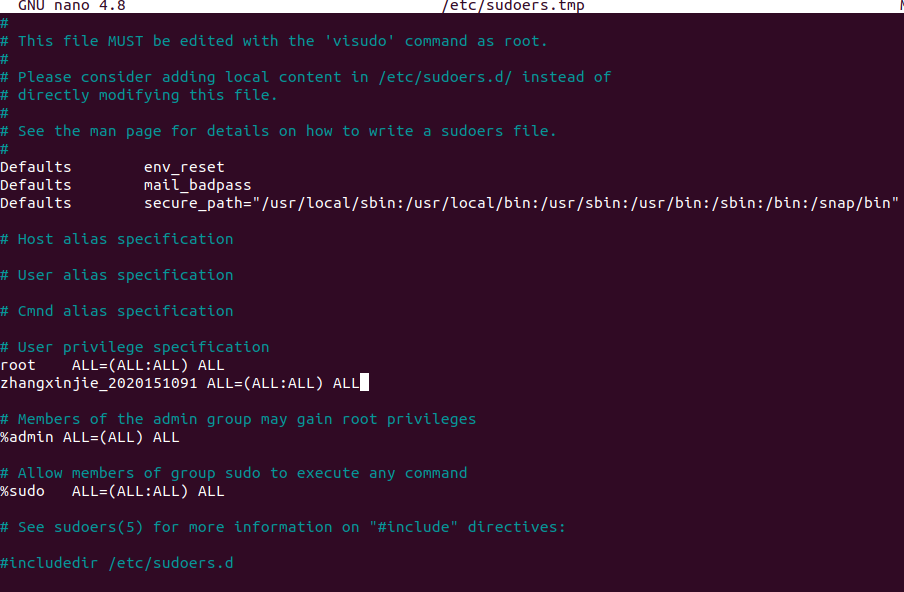
* 1. 注销当前用户，并以新建的用户身份登录，登录后运行 $ whoami，并进行截图；



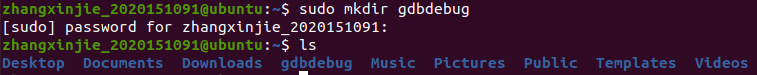
1. 新建用户主目录下创建子目录：**gdbdebug**, 并进入gdbdebug 子目录。将过程和结果截图。（10分）

2.1 使用指令*$sudo mkdir gdbbug*创建子目录，此时发现报错，报错信息为：“not in the sudoers file”，经过网上查询，发现新用户不具有sudo的权限，因此我们需要给当前用户sudo权限。

2.2 使用指令*$su*有再输入*visudo*进入root，在# User privilege specification栏目下边添加此账户的sudo权限



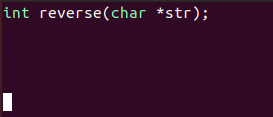
2.3 重新进入控制台，再次使用指令*#sudo mkdir gdbdebug*，成功创建，再输入*$ls*展示子目录如下



2.4 输入指令*$cd gdbdebug*进入子目录文件夹，再输入*$vi reverse.h*进入vi编辑器

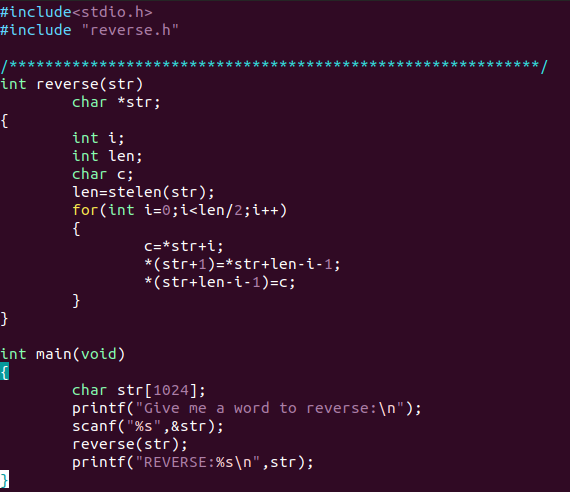


1. 使用vi编辑以下两个文件并编译和运行，截图（30分）
   1. 编辑reverse.h



在声明函数之后点击Esc退出编辑，并用:wq保存编辑的内容，但是在输入:wq之后报错，声明为”E212:Can’t open file for writing.”，通过查询资料得知是因为权限不足导致，退出到控制开输入指令*$su*进入超级用户模式，重新编辑reverse.h，再次输入:wq，此时成功保存。

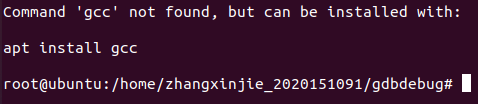
* 1. 编辑reverse.c



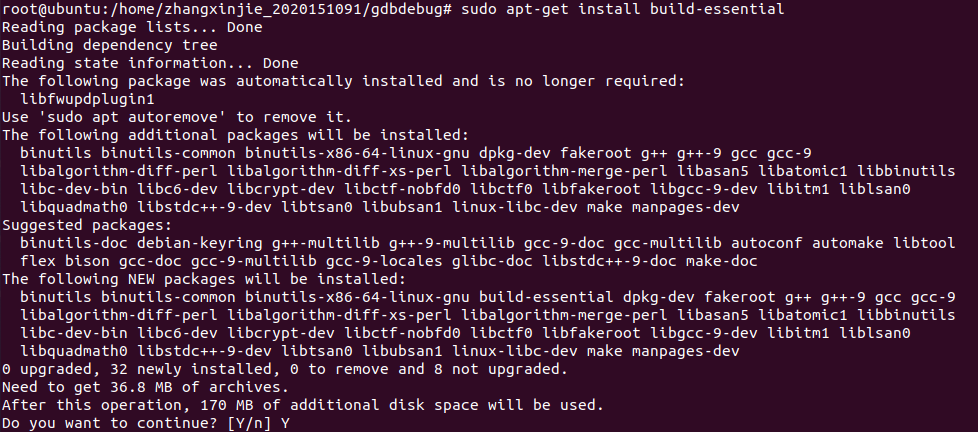
* 1. 按以下步骤编译，如有警告信息，请修改代码至无警告信息

$gcc –Wall reverse.c –o reverse

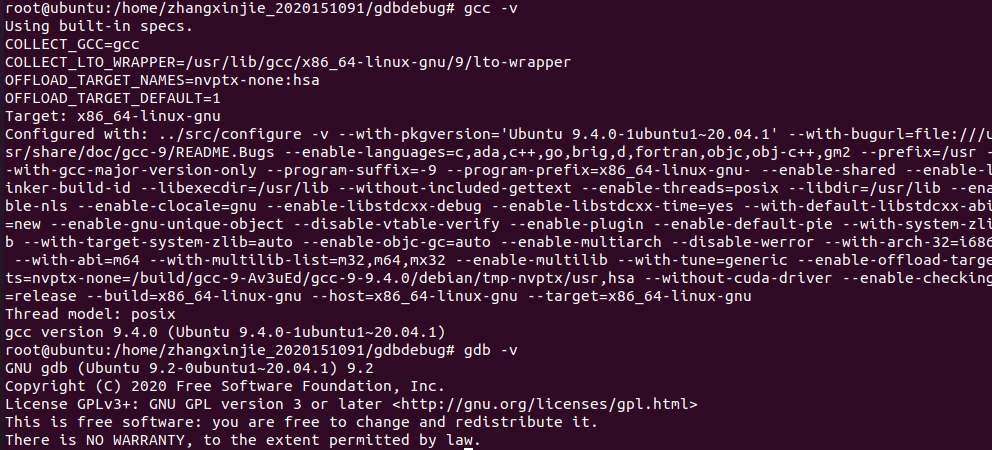
输入上述指令，发现报错，是因为未安装gcc工具链导致。



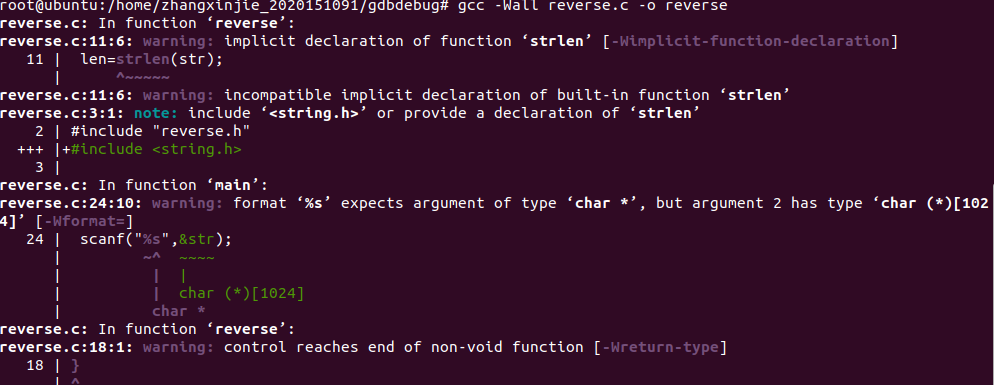
解决方法：输入命令*#sudo apt-get install build-essential*即可安装gcc、g++、gdb等一整套工具链。



分别输入指令gcc -v和gdb -v查看是否安装成功



再次输入$gcc –Wall reverse.c –o reverse，代码出现如下错误提示



错误提示1是因为函数strlen没有声明，解决方法：在头文件添加#include<string.h>

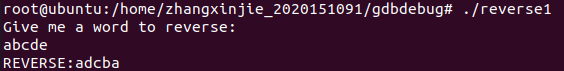
错误提示2是因为在main函数中%s的输入格式不对，应去掉取地址符

错误提示3是因为函数没有返回值，在函数末尾添加return 1即可。

在主函数添加完成后，再次输入指令尝试编译，没有了报错信息，说明成功编译。

* 1. 运行程序

$./reverse

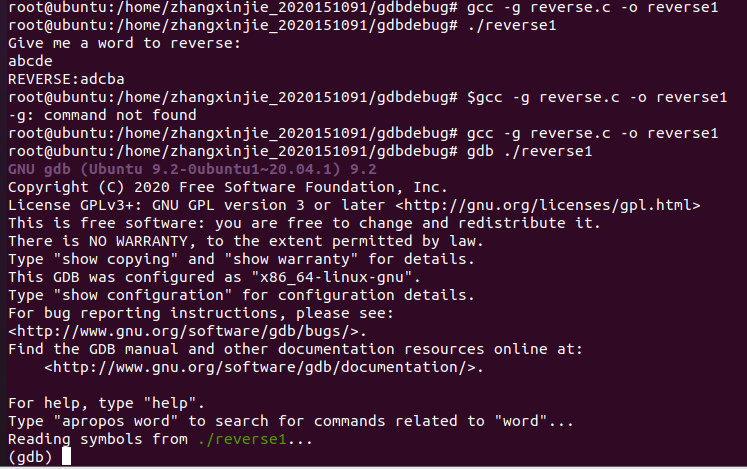


虽然程序成功执行，但是我们发现结果是错误的。

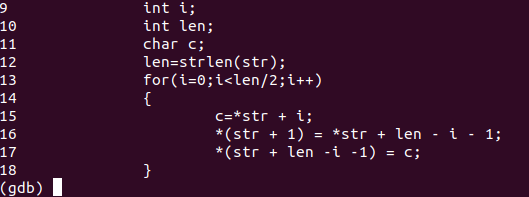
1. 按照以下过程调试并修正reverse.c,请参考过程截图。（30分）
   1. 编译时加入调试信息

**$gcc -g reverse.c -o reverse1**

* 1. 启用GDB调试

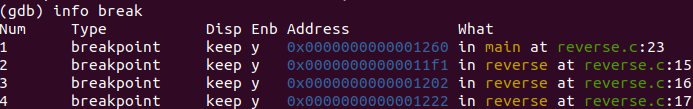


* 1. 键入list,查看源代码并根据行号/函数名设置断点（断点位置仅供参考，不需雷同



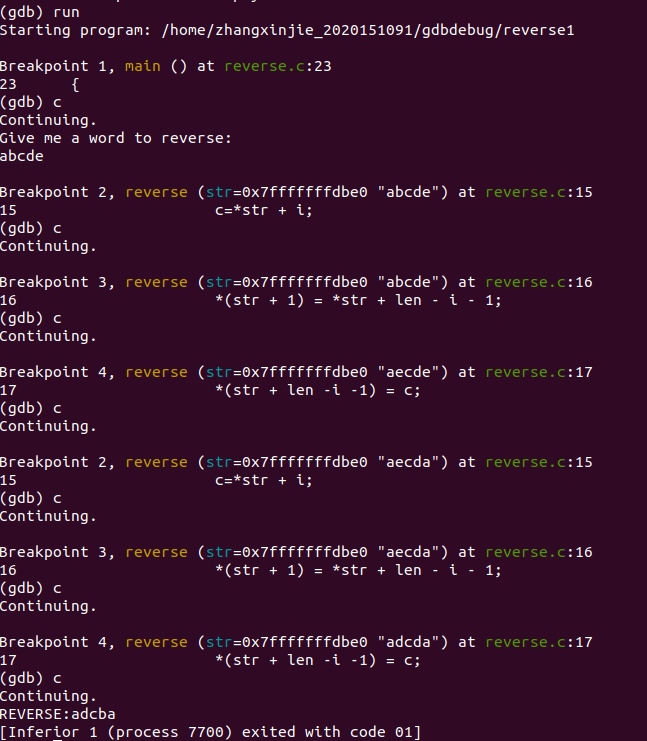
由于函数在for循环中出现错误的可能性较大，因此在15、16、17行设置断点。

设置完断点之后，输入指令*info break*即可查看设置的断点



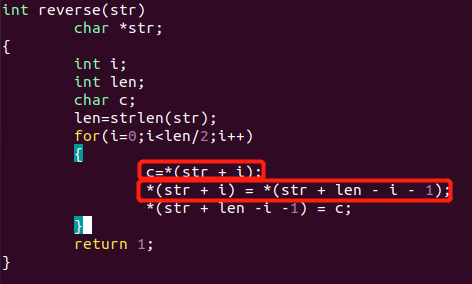
* 1. 观察变量值，并作分析，推测错误（过程仅供参考，不需雷同）

输入run指令之后进行gdb调试，可以键入c跳过当前断点。

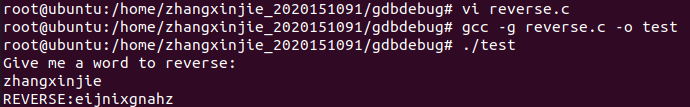


通过数据比较可以发现，在第一次操作中，本应该被e替换的a却没有被替换，而第二位的b却被替换了，通过代码分析得知，在字符替换过程中，本应通过指针交换首位和末位的字符，但在运行过程中，指针的位置却发生了偏移，因此在源代码中，\*str+i应该改为\*(str+i)，这是因为在C语言的运算优先级中，指针的运算优先级是高于加法的，因此在此处需要加入括号。Vi

* 1. 修正程序并运行

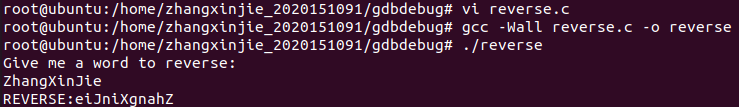


修改完成之后重新编译运行。



结果正确无误。

**四、实验结果**



结果如程序运行所示，通过比较，程序成功运行，输出结果也是正确的。

**五、实验总结与体会**

在本次实验中，遇到了几个问题，第一个是权限不足问题，在进入root之后在超级用户的栏目下添加自己的账户，在保存之后，能够顺利进行自己所需要的操作；第二个问题实在编辑“reverse.h“文件的时候，出现了点击上下左右方向键的时候会出现自动换行和a、b、c、d四个字符，通过查询资料得知，该问题是vim版本的原因，通过输入指令*$apt install vim*重新安装vim之后成功解决问题。

实验体会：通过此次实验，我初步了解了Linux操作系统，并初步掌握了在Linux操作系统下的常用命令。学会了gcc编译工具的基本用法，同时也学习了如何对代码进行gdb调试。

|  |
| --- |
| **指导教师批阅意见：**  **成绩评定：**  指导教师签字：  年 月 日 |
| 备注： |