实验四 GCC 和 GDB 的使用

小组成员： 陆烽 指导教师：周庆国

一、实验目的：

1. 掌握Linux操作系统下最常用的C语言编译器gcc的使用
2. 掌握Linux操作系统下最常用的代码调试器gdb的使用
3. 掌握调试代码的基本方法，如观察变量、设置断点等。

二、实验内容：

阅读在线帮助命令man gcc，以及man gdb的内容,了解gcc、gdb的基本使用。使用gdb观察变量、设置断点等。

三、实验步骤：

1. gcc编译helloworld文档

* 编译： gcc helloworld.c –o helloworld
* 运行： ./helloworld

2.分布解析gcc编译原理并查看相关文档属性

hofrat@rtl26:~/C$ gcc -E hello.c -o hello.i

hofrat@rtl26:~/C$ ls

hello.c hello.i

hofrat@rtl26:~/C$ gcc -S hello.i -o hello.s

hofrat@rtl26:~/C$ ls

hello.c hello.i hello.s

hofrat@rtl26:~/C$ gcc -c hello.s -o hello.o

hofrat@rtl26:~/C$ ls

hello.c hello.i hello.o hello.s

hofrat@rtl26:~/C$ gcc hello.o -o hello

hofrat@rtl26:~/C$ ls

hello hello.c hello.i hello.o hello.s

hofrat@rtl26:~/C$ ./hello

hello, world!

查看文件属性

hofrat@rtl26:~/C$ ls -l

total 36

-rwxr-xr-x 1 hofrat hofrat 6425 2010-05-28 07:32 hello

-rw-r--r-- 1 hofrat hofrat 126 2010-05-28 07:16 hello.c

-rw-r--r-- 1 hofrat hofrat 14481 2010-05-28 07:30 hello.i

-rw-r--r-- 1 hofrat hofrat 860 2010-05-28 07:31 hello.o

-rw-r--r-- 1 hofrat hofrat 431 2010-05-28 07:31 hello.s

3.GDB调试

命令用例

* gcc –g test.c //-g表示编译过程中生成可调式程序，默认生成a.out
* gdb //运行gdb
* (gdb) file a.out //打开a.out文件
* (gdb) list //显示代码
* (gdb) break 5 //设置断点在第5行
* (gdb) run //运行程序
* (gdb) print h //打印变量h值
* (gdb) continue //继续执行
* (gdb) quit //退出gdb
* (gdb) info break //查看断点信息
* (gdb) next（或n） //单条语句执行，next命令简写
* (gdb) backtrace（或bt） //查看各级函数调用及参数
* (gdb) step（s）//执行下一行语句，有函数调用则进入到函数中
* (gdb) set var //修改变量的值
* (gdb) finish //退出函数

四、收获总结。

初步学习了gcc编译的原理和过程，了解了其各步骤生成的文件信息，

了解了如何利用GDB对程序进行调试，其主要功能有：

* + 1、启动你的程序，可以按照你的自定义的要求随心所欲的运行程序。
  + 2、可让被调试的程序在你所指定的调置的断点处停住。（断点可以是条件表达式）
  + 3、当程序被停住时，可以检查此时你的程序中所发生的事。
  + 4、动态的改变你程序的执行环境。
* gdb的主要用途 – 修复bug；分析程序结构。