**实验一 逻辑及位操作运算**

**一、实验目的**

1. 掌握逻辑运算及位操作运算的工作原理；
2. 熟悉gcc、objdump、gdb工具的使用方法。

**二、实验环境**

PC机，Linux 32-bit i386，C/汇编语言

gcc、objdump、gdb

**三、实验学时**

2学时

**四、实验内容**

4.1 实验内容一

4.1.1 代码及编程思路

代码：

#include <stdio.h>

int main(){

printf("hello,world\n");

}

编程思路：

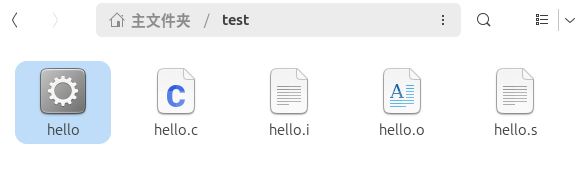
无

4.1.2 实验步骤

1. 编写程序
2. 使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件
3. 运行可执行文件

4.1.3 程序运行截屏





4.1.4 程出现问题及解决办法

无

4.2 实验内容二

4.2.1 代码及编程思路

代码：

#include <stdio.h>

void main(){

int x=3,y=5,z;

z=x+y;

printf("z=%d\n",z);

return;

}

编程思路：

无

4.2.2 实验步骤

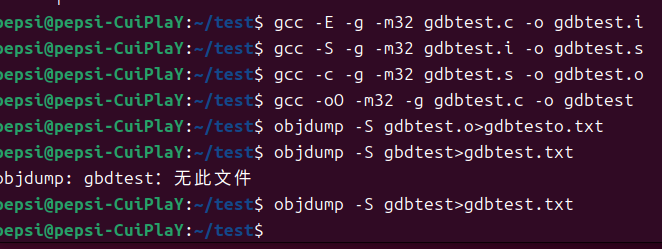
（1）编写程序

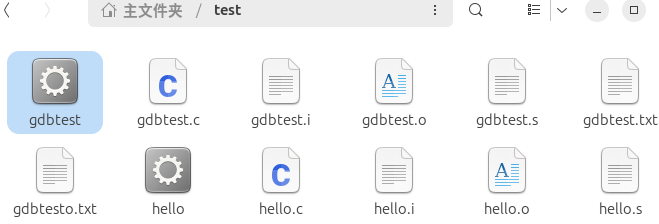
（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

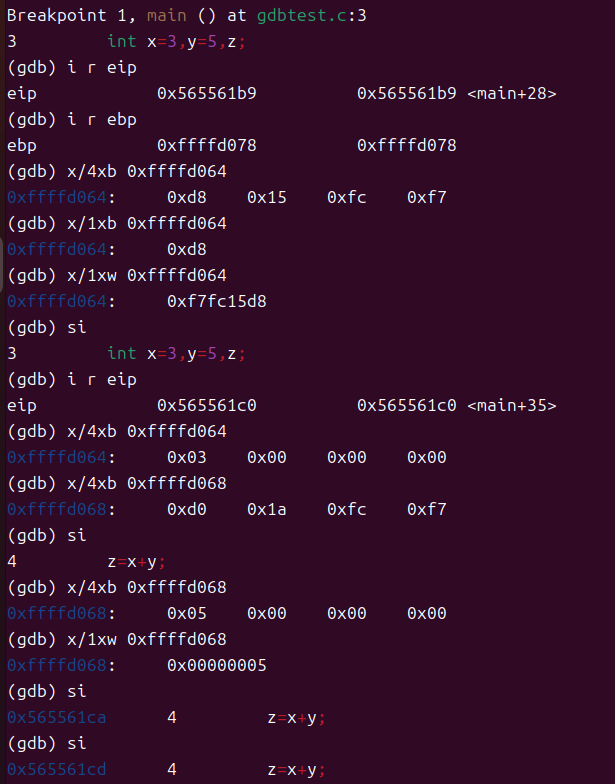
（3）使用objdump指令进行反汇编

（4）使用gdb指令对程序进行调试

4.2.3 程序运行截屏







4.2.4 程出现问题及解决办法

在使用指令过程中可能会出现找不到文件，应该仔细查看文件名是否正确。

4.3 实验内容三

4.3.1 代码及编程思路

代码：

#include <stdio.h>

void main(){

int a=5;

unsigned int b=3;

short c=5;

int d=0;

a=~a;

b=~b;

c=~c;

d=a&b;

d=a^b;

d=a|b;

return;

}

编程思路：

通过简单的位运算符来理解C语言中的位运算

4.3.2 实验步骤

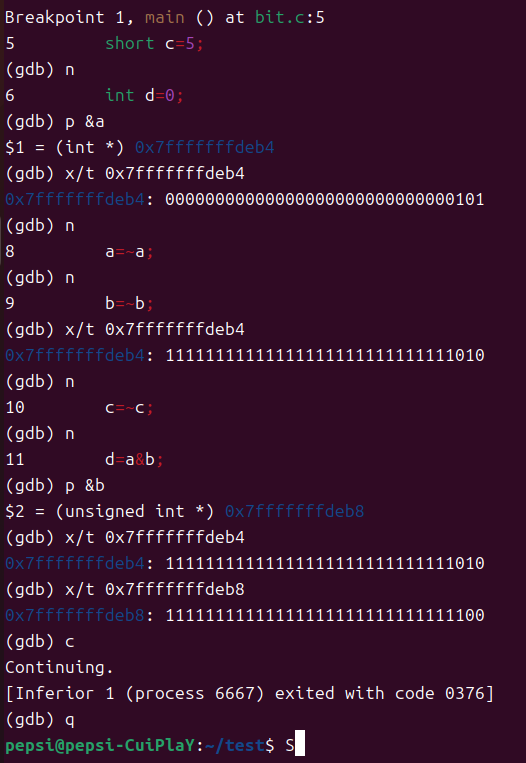
（1）编写程序

（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

（3）使用objdump指令进行反汇编

（4）使用gdb指令对程序进行调试，查看结果

4.3.3 程序运行截屏



4.3.4 程出现问题及解决办法

无

4.4 实验内容四

4.4.1 代码及编程思路

代码：

#include <stdio.h>

void main(){

int a=5;

unsigned int b=5;

short c=5;

int d=0;

a=!a;

b=!b;

c=!c;

d=a&&b;

d=a||b;

return;

}

编程思路：

通过简单的逻辑运算符来理解C语言中的逻辑运算

4.4.2 实验步骤

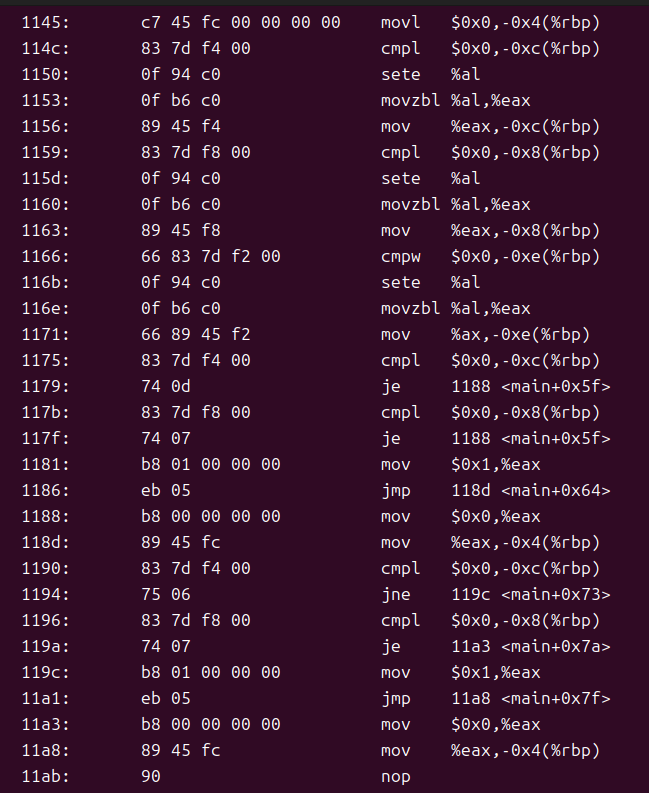
实验步骤：

（1）编写程序

（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

（3）使用objdump指令进行反汇编，查看反汇编对应的结果

4.4.3 程序运行截屏



4.4.4 程出现问题及解决办法

无

4.5 实验内容五

4.5.1 代码及编程思路

代码：

#include <stdio.h>

void main(){

int a=0x80000000;

unsigned int b=0x80000000;

short c=0x8000;

unsigned short d=0x8000;

a=a>>4;

b=b>>4;

a=c;

a=d;

b=c;

b=d;

return;

}

编程思路：

通过简单的移位运算符来理解C语言中的移位运算

4.5.2 实验步骤

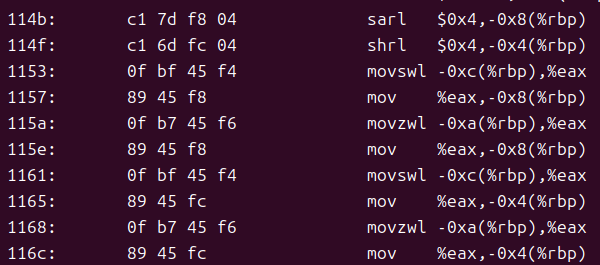
（1）编写程序

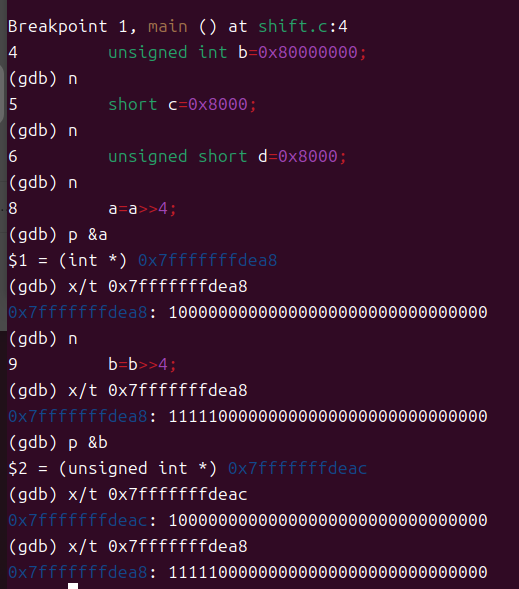
（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

（3）使用objdump指令进行反汇编

（4）使用gdb指令对程序进行调试，查看结果

4.5.3 程序运行截屏





4.5.4 程出现问题及解决办法

无

4.6 实验内容六

4.6.1 代码及编程思路

代码：

#include <stdio.h>

void main(){

int a=3,b=89;

printf("before swapping: a=%d,b=%d\n",a,b);

a=a^b;

b=b^a;

a=a^b;

printf("after swapping: a=%d,b=%d\n",a,b);

return;

}

编程思路：

通过简单的异或运算符来理解C语言中两个变量的值

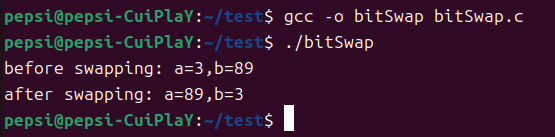
4.6.2 实验步骤

（1）编写程序

（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

（3）执行程序

4.6.3 程序运行截屏



4.6.4 程出现问题及解决办法

无

**五、实验收获**

吐槽一下实验过程中的心路，如：通过实验，提醒自己以后编程要注意什么

在编程实验里，用GCC和GDB来调试代码，还有搞明白位运算符和逻辑运算符怎样工作，这整个过程能学到不少东西，也确实有不少坑。用GCC编译C语言程序时，-g选项一定要加，这样GDB才能进行源码级调试。每次编码的时候都留心这些小细节，长期下来，代码质量肯定会大幅提升，debug的时候也会省好多事儿，减少很多头发掉落。碰到位运算和逻辑运算，一定得搞清楚各个运算符之间的优先级，不然容易出错。如果表达式复杂，记得用括号明确一下运算顺序，这样不仅能避免错误，还能提高代码的清晰度。