**实验二 算术运算**

**一、实验目的**

1. 掌握算术运算的工作原理
2. 掌握数的机器级表示及转换
3. 掌握整数加减运算及其对标志位的影响
4. 掌握浮点数的表示和基本运算

**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()

{

int a=0x987654321,b=0x87654321,c,d;

unsigned int ua=0x987654321,ub=0x87654321,uc,ud;

c=a+b;

uc=ua+ub;

d=a-b;

ud=ua-ub;

printf("%d+(%d)=%d\n",a,b,c);

printf("%u+(%u)=%u\n",ua,ub,uc);

printf("%d-(%d)=%d\n",a,b,d);

printf("%u-(%u)=%u\n",ua,ub,ud);

}

编程思路：

通过带符号与无符号的整数加减法理解加分器的原理

4.1.2 实验步骤

（1）编写程序

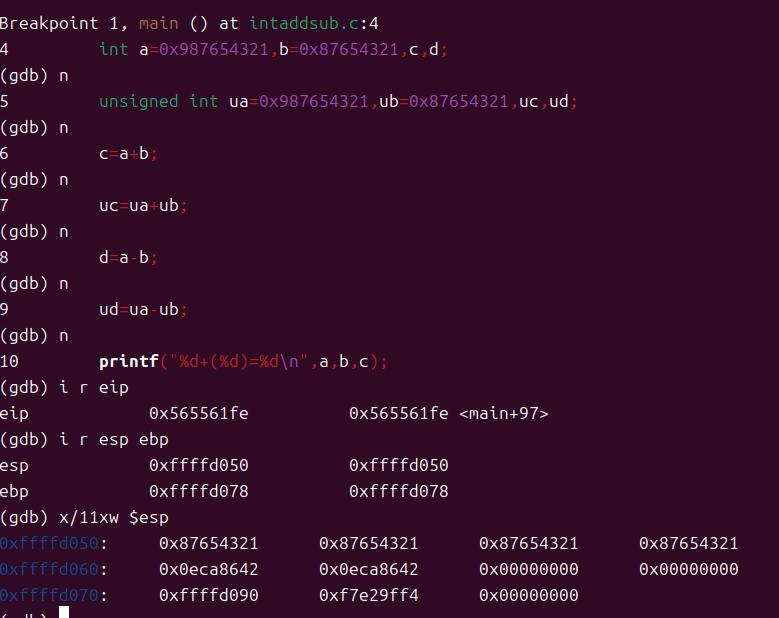
（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

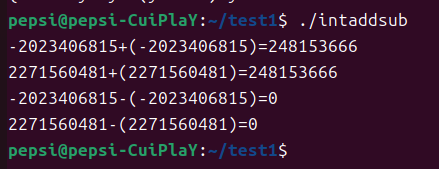
（3）使用objdump指令进行反汇编

（4）使用gdb指令对程序进行调试

（5）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

#include <stdio.h>

int main()

{

int a=0x987654321,b=0x87654321,c,d;

unsigned int ua=0x987654321,ub=0x87654321,uc,ud;

c=a+b;

uc=ua+ub;

d=a-b;

ud=ua-ub;

printf("%d+(%d)=%d\n",a,b,c);

printf("%u+(%u)=%u\n",ua,ub,uc);

printf("%d-(%d)=%d\n",a,b,d);

printf("%u-(%u)=%u\n",ua,ub,ud);

}

编程思路：

通过带符号与无符号的整数加减法理解加分器的OF,ZF,SF,CF标志位

4.2.2 实验步骤

（1）编写程序

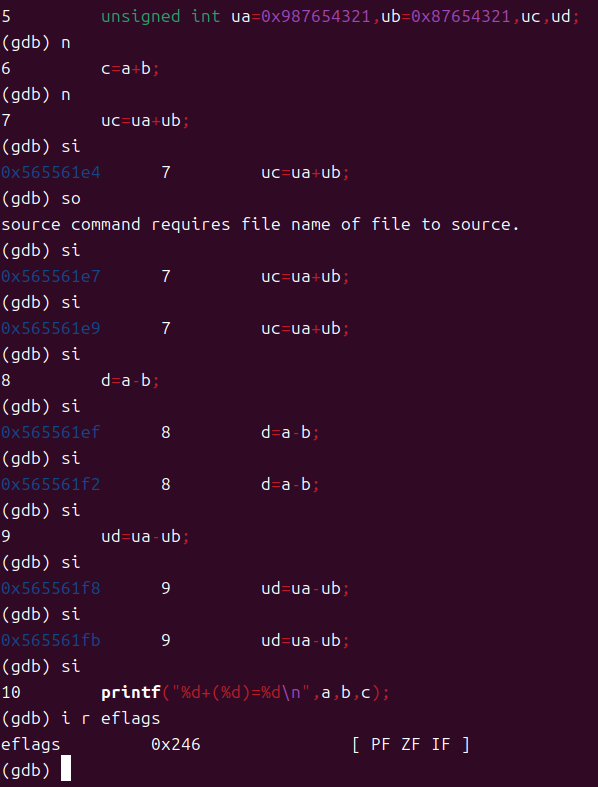
（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

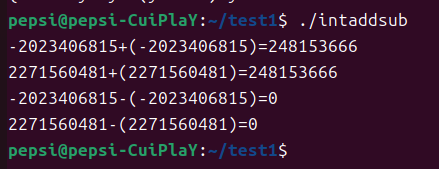
（3）使用objdump指令进行反汇编

（4）使用gdb指令对程序进行调试

（5）运行可执行文件

4.2.3 程序运行截屏





4.2.4 程出现问题及解决办法

无

4.3 实验内容三

4.3.1 代码及编程思路

代码：

#include <stdio.h>

int main()

{

unsigned ua=0xffffffff,ub=1,uc,ud;

int a=0x7ffffff0,b=0x7fffffff,c=-0x7ffffff0;

int d,e;

uc=ua+ub;

ud=ub-ua;

d=a+b;

e=c-b;

printf("%u+%u=%u\n",ua,ub,uc);

printf("%u-%u=%u\n",ub,ua,ud);

printf("%d+%d=%d\n",a,b,d);

printf("%d-%d=%d\n",c,b,e);

}

编程思路：

通过带符号与无符号的整数加减法理解运算结果的溢出问题

4.3.2 实验步骤

（1）编写程序

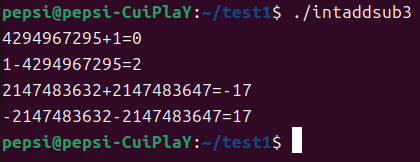
（2）使用指令分别对源程序进行预处理、编译、汇编和链接，最后生成可执行文件

（3）使用objdump指令进行反汇编

（4）使用gdb指令对程序进行调试

（5）运行可执行文件

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()

{

float finf1=4e38, finf2=5e38,finf3=6e38;//正无穷大

float fninf1 =- 4e38,fninf2 =- 5e38,fninf3 =- 6e38;//负无穷大

float fzero=0,fnzero =- fzero;

float fnormal1=5.0,fnormal2=0.1,fnnormal1 =- 5,fnnormal2 =- 0.1;//

float ffrac=1e-40,fnfrac =- 1e-40;

float fnan1=finf1+fninf1,fnan2 =- fnan1;

float finf4=fnormal1/fzero;

printf("%f%f%f\n",finf1,finf2,finf3);

printf("%f%f %f\n",fninf1,fninf2,fninf3);

printf("%f%f\n",fzero,fnzero);

printf("%f%.20f\n%f %20f\n",fnormal1,fnormal2,fnnormal1,fnnormal2);

printf("%.50f\n%.50f\n",ffrac,fnfrac);

printf("%f%f\n",fnan1,fnan2);

printf("%f\n",finf4);

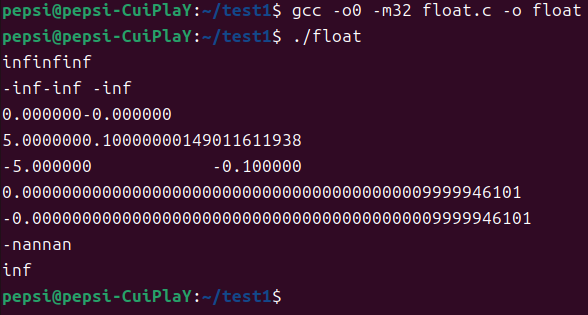
}

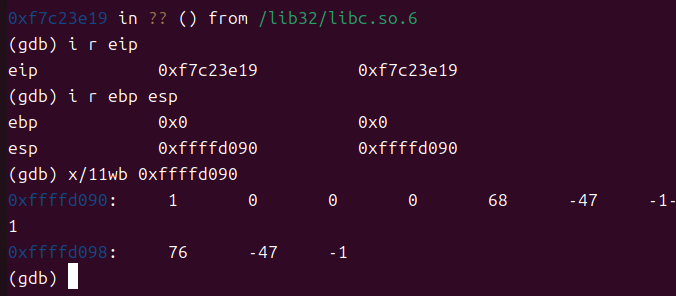
编程思路：

理解浮点数的表示

4.4.2 实验步骤

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

#include "stdio.h"

void main(){

double xd=0.1;

float xf=0.1;

int xi1=0x0ccccccd;

int xi2=0x0ccccd;

int xi3=0x0ccccc;

double yf, yi1,yi2,yi3;

yf=(xf-xd)\*100\*60\*60\*10\*2000;

yi1=((double)(xi1)/0x80000000-xd)\*100\*60\*60\*10\*2000;

yi2=((double)(xi2)/0x800000-xd)\*100\*60\*60\*10\*2000;

yi3=(xd-(double)(xi3)/0x800000)\*100\*60\*60\*10\*2000;

printf("%f //32bit/float\n",yf);

printf("%f //32bit / fixed point decimal / enter\n",yi1);

printf("%f //24bit/ fixed point decimal/enter\n",yi2);

printf("%f //24bit/ fixed point decimal/ abandon\n",yi3);

}

编码思路：

模拟爱国者导弹的程序

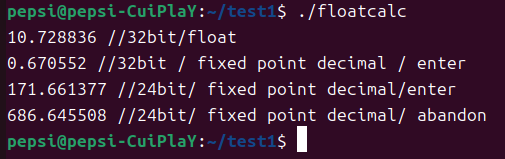
4.5.2 实验步骤

（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）运行可执行文件

4.5.3 程序运行截屏



4.5.4 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

细节决定成败。在进行算术运算时，一定要注意数据类型的转换和溢出问题。例如，在进行整数加减运算时，要注意判断结果是否超出了数据类型的范围，避免出现错误的结果。对于浮点数的表示和运算，要了解其原理和特点。浮点数的表示包括符号位、指数位和尾数位，要熟悉这些部分的作用和计算方法。在进行浮点数运算时，要注意精度问题，避免因为精度损失导致的错误结果。