**6**

1. R[ebp]+8;R[ebp]+12;

R[ebp]+16;

(2)void func(int \*xptr, int \*yptr, int \*zptr)

{

int a=\*yptr;

int b=\*zptu;

int c=\*xptu;

\*yptr=c;

\*zptu=a;

\*xptu=b;

}

**7**

1. R[edx]=x
2. R[edx]=x+y+4
3. R[edx]=x+y\*8
4. R[edx]=y+x\*2+12
5. R[edx]=4\*y
6. R[edx]=x+y

**9**

1.将y送ECX

2.将y\*256送ECX

3.将x送EAX

4.将k送EDX

5.将x\*k送EAX

6.将z送EDX

7.将z&0xFFF0送EDX

8.将z&0xFFF0+y\*256送EDX

9.将x\*k-(z&0xFFF0+y\*256)送EAX

缺失部分int v= x\*k-(z&0xFFF0+y\*256) ;

**12**

(1)

1.将x送DL

2.将p送EAX

3.判断p是否为0

4.若p为0，执行.L1

5.判断x的第一位是否为0

6.若x>=0，则执行.L1

7.\*p+=x

8. .L1:

C语言if语句中的条件表达式可以对多个条件进行逻辑运算，而汇编代码中一条指令只能进行一种逻辑运算。

(2)

1 void comp(char x, int \*p)

2 {

3 if (p!=0)

4 if (x<0)

5 \*p+=x;

6 }

**13**

x\*y

x<=-100

y>x

x+y

x-y

x>=16

x&y

**14**

（1）

1.将x送BX

2.将y送SI

3.将k送CX

4..L1:

5.将y送DX

6.将y送AX

7.将y的符号扩展16位送DX

8.将y%k送DX

9.将x\*(y%k) 送BX

10.将k-1 送CX

11.得OF=CF=0，负数则SF=1，零则ZF=1

12.若k小于等于0，则转.L2

13.将y与k相减得到各标志

14.若y大于k，则转.L1

15..L2:

16.将x\*(y%k)送AX

（2）被调用者保存寄存器BX、SI，调用者保存寄存器AX、CX、DX。

在该函数过程体前面的准备阶段，被调用者保存的寄存器EBX和ESI必须保存到栈中。

（3）因为执行第8行除法指令前必须先将被除数扩展为32位。

**15**

x!=0

y^=x

x>>=1

y&0x1

函数f1的功能：当x中有奇数个1，则返回为1，否则返回0。

**16**

当x+3>7时转标号.L7处执行，否则，按照跳转表中的地址转移执行，x与跳转目标处标号的关系如下：

x+3=0：.L7

x+3=1：.L2

x+3=2：.L2

x+3=3：.L3

x+3=4：.L4

x+3=5：.L5

x+3=6：.L7

x+3=7：.L6

**17**

unsigned int test(char a, unsigned short b, unsigned short c, short \*p);

**19**

x==0

0

x>>1

(x & 0x1) + rv

该函数的功能为计算x的各个数位中1的个数。

**25**

（1）node所需存储空间有16字节。成员p、s.x、s.y和next的偏移地址分别为0、4、8、12。

（2）

np->s.y

&(np->s.x)

np

**29**

（1）（2）执行第7行和第10行指令后栈中的信息存放情况如下图所示。

执行第7行后的栈

ESP

EBP

EBP在P中旧值

08 04 85 c8

返回P的地址

bf fc 08 00

00 00 00 08

00 00 00 10

00 00 00 05

被调用者保存寄存器在P中的旧值

buf[7]~ buf[4]

buf[3]~ buf[0]

执行第10行后的栈

ESP

EBP

EBP在P中旧值

08 41 39 38

返回P的地址

37 36 35 34

33 32 31 30

46 45 44 43

42 41 39 38

被调用者保存寄存器在P中的旧值

buf[7]~ buf[4]

buf[3]~ buf[0]

33 32 31 30

37 36 35 34

gets入口参数

bf fc 07 d8

（3）当执行到getline的ret指令时，发生段错误是因为执行getline的ret指令时得到的返回地址为0x8413938，这个地址所在存储段可能是不可执行的数据段。

（4）EBX、ESI、EBP和EDI在P中的内容已被破坏。

（5）malloc函数的参数应该为strlen(buf)+1。