1. 假设变量*x*和*y*分别存放在寄存器EAX和ECX中，请给出以下每条指令执行后寄存器EDX中的结果。

（1）leal (%eax), %edx

（2）leal 4(%eax, %ecx), %edx

（3）leal (%eax, %ecx, 8), %edx

（4）leal 0xC(%ecx, %eax, 2), %edx

（5）leal ( , %eax, 4), %edx

（6）leal (%eax, %ecx), %edx

1. x
2. x+y+4
3. x+8y
4. y+2x+12
5. 4x
6. x+y
7. 假设函数operate的部分C代码如下：

1 int operate(int x, int y, int z, int k)

2 {

3 int v = x \* k – (z & 65220 + y << 8) ;

4 return v;

5 }

以下汇编代码用来实现第3行语句的功能，请根据以下汇编代码，填写operate函数缺失的部分。

1 movl 12(%ebp), %ecx

2 sall（左移）$8, %ecx

3 movl 8(%ebp), %eax

4 movl 20(%ebp), %edx

5 imull %edx, %eax

6 movl 16(%ebp), %edx

7 andl $65520, %edx

8 addl %ecx, %edx

9 subl %edx, %eax

1. 已知函数func的C语言代码框架及其过程体对应的汇编代码如下图所示，根据对应的汇编代码填写C代码中缺失的表达式。

1 int func(int x, int y)

2 {

3 int z = x \* y ;

4 if ( x <= -100 ) {

5 if ( x < y )

6 z = x + y ;

7 else

8 z = x - y ;

9 } else if ( x >= 16 )

10 z = x & y ;

11 return z;

12 }

1 movl 8(%ebp), %eax

2 movl 12(%ebp)(x), %edx(y)

3 cmpl $-100, %eax

4 jg(>) .L1

5 cmpl %eax, %edx

6 jle(<=) .L2

7 addl %edx, %eax

8 jmp .L3

9 .L2:

10 subl %edx, %eax

11 jmp .L3

12 .L1:

13 cmpl $16, %eax

14 jl(<) .L4

15 andl %edx, %eax

16 jmp .L3

17 .L4:

18 imull %edx, %eax

19 .L3:

1. 已知函数funct的C语言代码如下：

1 int funct(viod) {

2 int x, y;

3 scanf(“%x %x”, &x, &y);

4 return x-y;

5 }

函数funct对应的汇编代码如下：

1 funct：

2 push l %ebp

3 movl %esp, %ebp

4 subl $40, %esp

5 leal -8(%ebp), %eax

6 movl %eax, 8(%esp)

7 leal -4(%ebp), %eax

8 movl %eax, 4(%esp)

9 movl $.LC0, (%esp) //将指向字符串“%x %x”的指针入栈

10 call scanf //假定scanf执行后x=15,y=20

11 movl -4(%ebp), %eax

12 subl -8(%ebp), %eax

13 leave

14 ret

假设函数funct开始执行时，R[esp]=0xbc000020，R[ebp]=0xbc000030，执行第10行call指令后，scanf从标准输入读入的值为0x16和0x100，指向字符串“%x %x”的指针为0x804c000。回答下列问题或完成下列任务。

（1）执行第3、10和13行的指令后，寄存器EBP中的内容分别是什么？

（2）执行第3、10和13行的指令后，寄存器ESP中的内容分别是什么？

（3）局部变量*x*和*y*所在存储单元的地址分别是什么？

（4）画出执行第10行指令后funct的栈帧，指出栈帧中的内容及其地址。

（1）3 0xbc00001C

10 0xbbffffec

13 0xbc000030

（2）3 0xbc00001C

10 0xbbffffec

13 0xbc000020

（3）x 0xbc000018

y 0xbc000014

（4）