1. 下列那一条指令是正确的： C 。

A movb $0xE, (%ebx) B movl %rax, (%rsp)

C movq %rax, %rdx D movw (%rax), 4(%rsp)

1. 1) mov (%eax, %eax, 4), %eax

2) lea (%eax, %eax, 4), %eax

上面指令中的那一条会产生如下结果: %eax = 5 \* %eax?( C )

A 1)和2)都不会 B 1) C 2) D 1)和2)都会

1. 下列指令中不会改变PC寄存器内容的是 A 。

A ADD B JMP C CALL D RET

1. 已知int P[M][N]和int Q[N][M]，有以下函数：

int addfun( int i,int j){

return P[i][j]+Q[j][i];}

对应有汇编代码如下，请问M和 N分别是多少？

addfun:

movl %edi, %edx

shl $2,%edx

addl %esi,%edx

movl %esi,%eax

shll $2,%eax

addl %eax,%edi

movl Q(,%rdi,4),%eax

addl P(,%rdx,4), %eax

ret

M=N=4

**5,** 已知函数Sum\_fun的C语言代码及其对应的x86-64汇编代码框架，请补齐缺失的汇编代码

void Sum\_fun (int \*p)

{

int a[4] = {1, 2, 4, 8}, i = 0;

do {

\*p += a[i];

i = i+1;

} while(i < 4)

}

subq $0x10, %rsp

movl $0x08, 12(%rsp)

movl $4, 8(%rsp) #

movl $2, 4(%rsp)

movl $1, (%rsp)

movl $0, %eax

movl $0, %ecx

.L1

movslq %eax, %rax

addl (%rsp,%rax, 4), %ecx #

addq $1, %rax #

cmpq $4, %rax #

jl .L1

addl %ecx, (%rdi)

addq $0x10, %rsp #

ret

6，考虑如下一个C函数和对应的x86-64汇编代码。C代码若有缺失，请在下划线处补全，并且填写下面的跳转表。

int s\_f (int a, int b)

{

40045c < s\_f >:

40045c: lea -0xd2(%rdi), %eax

400462: cmp $0x9, %eax

400465: ja 40048a < s\_f +0x2e>

400467: mov %eax, %eax

400469: jmpq \*0x400590(, %rax, 8)

400470: lea (%rsi,%rsi,2),%eax

400473: lea (%rsi,%rax,4),%eax

400476: retq

400477: mov $0x4743, %esi

40047c: mov %esi, %eax

40047e: imul %esi, %eax

400481: retq

400482: mov %esi, %eax

400484: sub %edi, %eax

400486: retq

400487: add $0xd, %esi

40048a: lea -0x9(%rsi), %eax

40048d: retq

switch(a)

{

case 210:

b \*= 13;

break

case 213:

b = 18243;

case 214:

b \*= b;

break

case 216:

case 218:

b -= a;

break

case 219:

b += 13;

提示：

0xd2 = 210

0x4743 = 18243

default:

b -= 9;

}

return b;

}

跳转表：

0x400590: \_0x400470 \_\_\_\_ 0x400598: \_\_\_\_0x40048a\_\_\_\_\_\_\_\_\_\_\_\_

0x4005a0: \_0x40048a \_\_\_\_\_ 0x4005a8: \_\_\_\_0x400477\_\_\_\_\_\_\_\_\_\_\_\_

0x4005b0: \_0x40047c \_\_\_\_\_0x4005b8: \_\_\_\_0x40048a\_\_\_\_\_\_\_\_\_\_\_\_

0x4005c0: \_0x400482 \_\_\_\_ 0x4005c8: \_\_\_\_\_0x40048a\_\_\_\_\_\_\_\_\_\_\_

0x4005d0: \_0x400482 \_\_\_\_\_ 0x4005d8: \_\_\_\_0x400487\_\_\_\_\_\_\_\_\_\_\_\_

1. 函数P与结构体test的定义如下，并请完成以下的问题：

struct test {

char \*a;

char b;

int c;

};

long P(long x, long y, struct test \*s)

{

long u = Q(y);

long v = Q(x);

return u+v+(s->c);

}

1. P:
2. pushq %rbp
3. pushq %rbx
4. pushq %r12
5. subq $8, %rsp
6. movq %rdx, %r12
7. movq %rdi , %rbp
8. movq %rsi , %rdi
9. call Q
10. movq %rax , %rbx
11. movq %rbp , %rdi
12. call Q
13. addq %rbx, %rax
14. addq 12(%r12) , %rax
15. addq $8, %rsp
16. popq %r12
17. popq %rbx
18. popq %rbp
19. ret

补全P函数对补全P函数对应的汇编代码中的缺失部分；