

# CS 24

## Introduction to Computing Systems

# x86-64 Mystery Programs

```
movzwl (%rdi),%eax
movzwl (%rsi),%edx
mov %dx,(%rdi)
mov %ax,(%rsi)
retq
```

0000000000000000 <f>:

```
0: 85 ff          test %edi,%edi
2: 7e 1b          jle 1f <f+0x1f>
4: 48 63 cf      movslq %edi,%rcx
7: 89 fa          mov %edi,%edx
9: b8 01 00 00 00 mov $0x1,%eax
e: 66 90          xchg %ax,%ax
10: 48 0f af c1   imul %rcx,%rax
14: 48 83 c1 ff   add $0xffffffffffffffff,%rcx
18: 48 83 c2 ff   add $0xffffffffffffffff,%rdx
1c: 75 f2          jne 10 <f+0x10>
1e: c3            retq
1f: b8 01 00 00 00 mov $0x1,%eax
24: c3            retq
```

0000000000401140 <f>:

401140: 50

401141: bf 0a 00 00 00

401146: e8 f5 fe ff ff

40114b: c7 00 05 00 00 00

401151: 59

401152: c3

**push** %rax

**mov** \$0xa,%edi

**callq** 401040 <malloc@plt>

**movl** \$0x5,(%rax)

**pop** %rcx

**retq**

0000000000000000 <f>:

0:	85 f6	test	%esi,%esi
2:	7e 18	jle	1c <f+0x1c>
4:	89 f0	mov	%esi,%eax
6:	31 c9	xor	%ecx,%ecx
8:	0f 1f 84 00 00 00 00	nopl	0x0(%rax,%rax,1)
f:	00		
10:	89 0c 8f	mov	%ecx,(%rdi,%rcx,4)
13:	48 83 c1 01	add	\$0x1,%rcx
17:	48 39 c8	cmp	%rcx,%rax
1a:	75 f4	jne	10 <f+0x10>
1c:	c3	retq	

```
callee:  
    mov  %edi, %eax  
    retq  
caller:  
    call callee  
    add  $1, %eax  
    retq
```

callee:

```
push %rbp
mov  %rsp, %rbp
mov  %edi, -4(%rbp)
mov  -4(%rbp), %eax
pop  %rbp
retq
```

caller:

```
push %rbp
mov  %rsp, %rbp
sub  $16, %rsp
mov  %edi, -4(%rbp)
mov  -4(%rbp), %edi
call callee
add  $1, %eax
add  $16, %rsp
pop  %rbp
retq
```