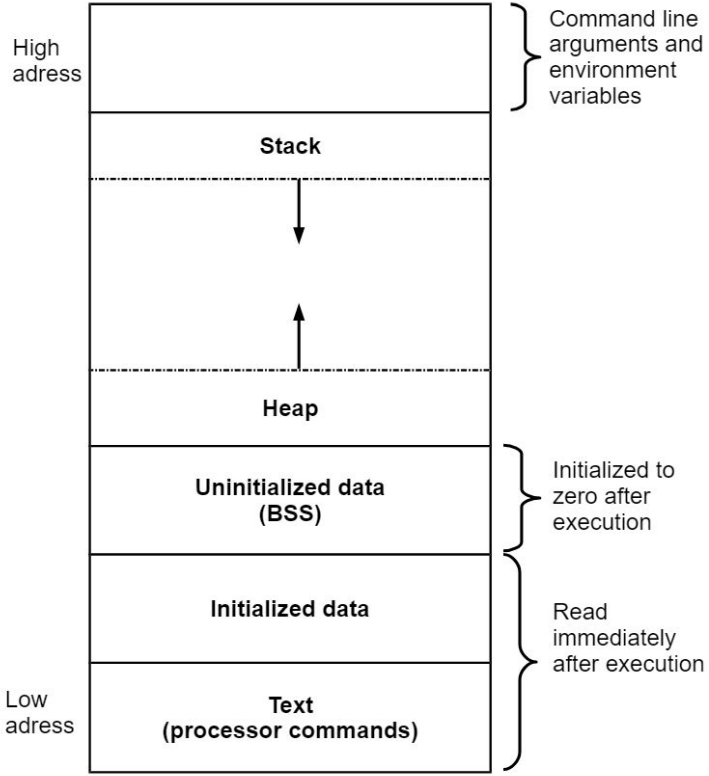
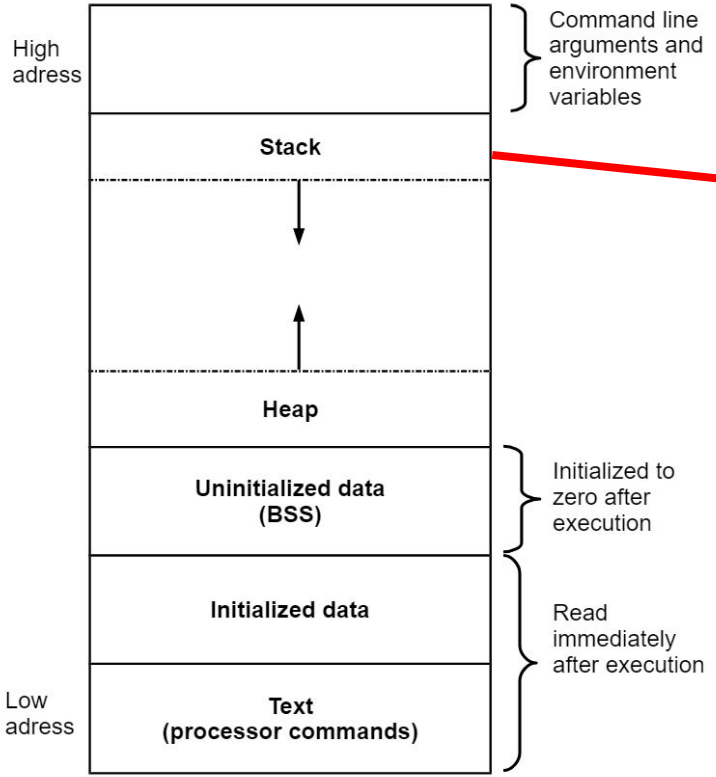
Recitation 7

Computer Architecture (section 1)

C Memory Layout

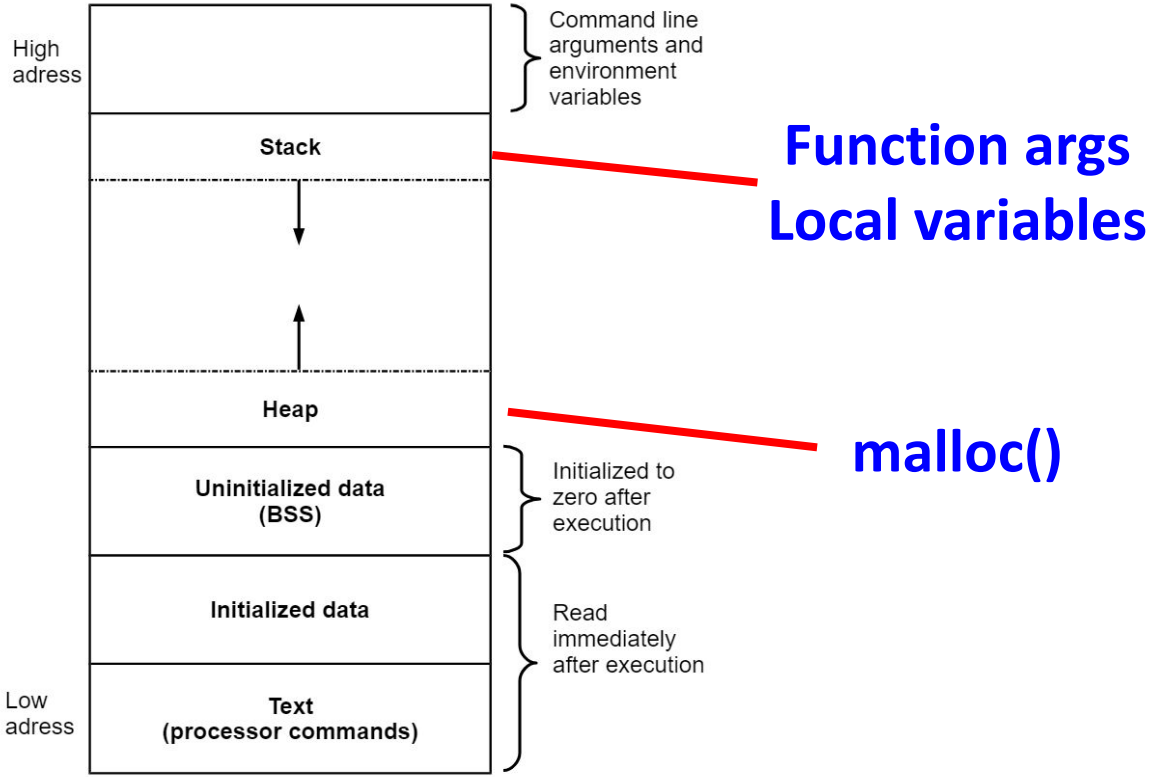


C Memory Layout

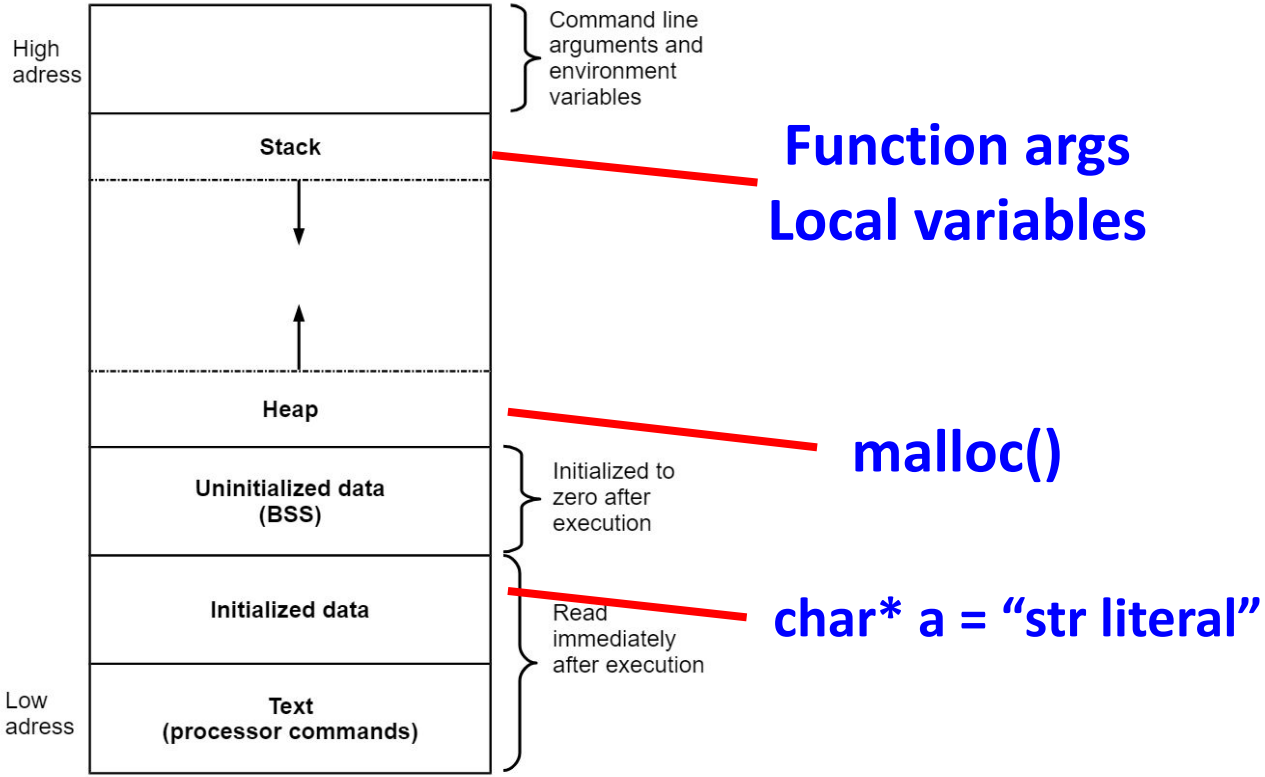


Function args
Local variables

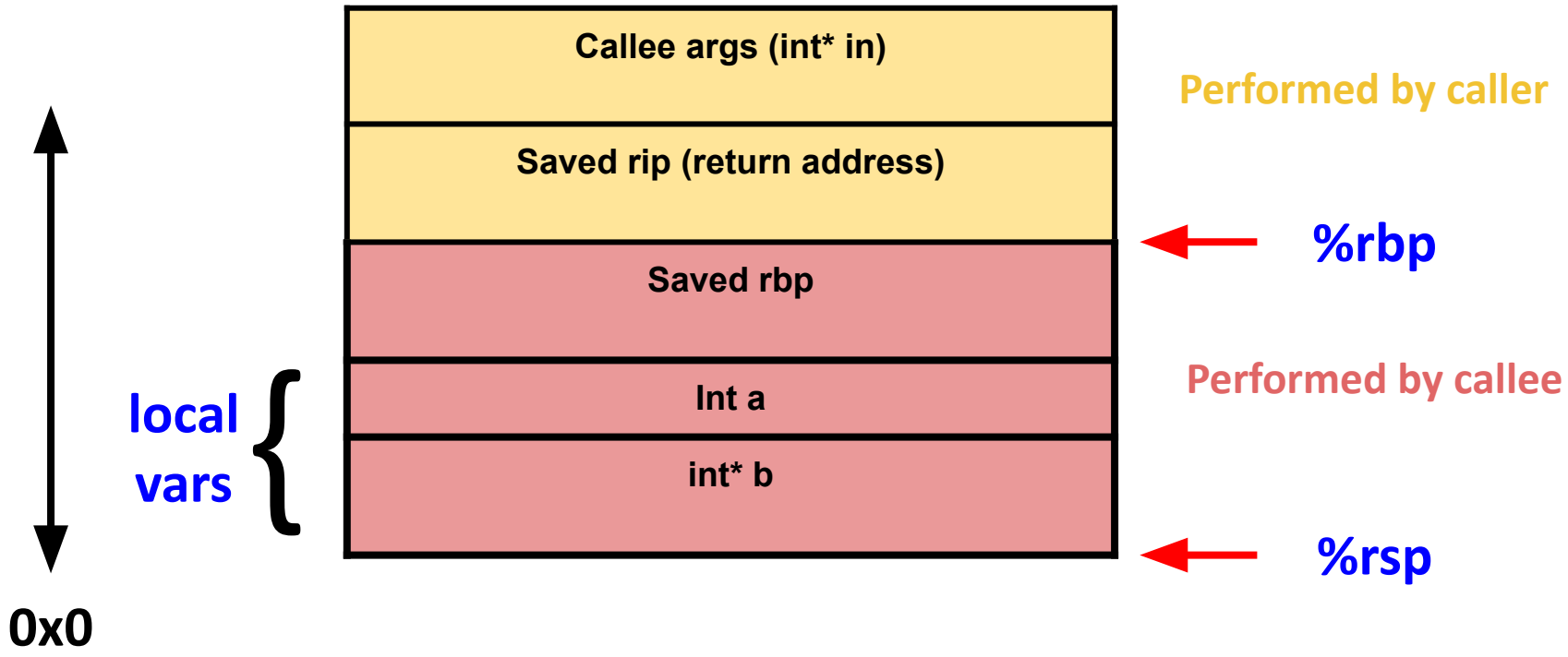
C Memory Layout



C Memory Layout



C Stack Frame



X86 procedure calls

- `call`
  - Push return address containing next instruction onto stack.
  - Set program counter to instruction after label.
- `ret`
  - Pop return address from the stack.
  - Set program counter to the return address.
- `leave`
  - High-level procedure exit.
  - Copies base pointer into stack pointer.
  - Restores old base pointer from stack.

X86 procedure calls

- `call`
  - Push return address containing next instruction onto stack.
  - Set program counter to instruction after label.
- `ret`
  - Pop return address from the stack.
  - Set program counter to the return address.
- `leave`
  - High-level procedure exit.
  - Copies base pointer into stack pointer.
  - Restores old base pointer from stack.

`enter` does the opposite, but is often avoided by compilers

x86-64 calling convention: Linux

Caller Arguments passed in:
%rdi
%rsi
%rdx
%rcx
%r8
%r9
Additional args passed on stack

Callee return value
%rax

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

CF	
PF	
ZF	
SF	
OF	

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

CF	
PF	
ZF	
SF	
OF	

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (32-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $4, %esp
        movl    12(%ebp), %eax
        movb    %al, -4(%ebp)
        cmpb    $122, -4(%ebp)
        jne     .L2
        movl    8(%ebp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        leave
        ret
main:
        pushl   %ebp
        movl    %esp, %ebp
        subl    $16, %esp
        movl    $100, -4(%ebp)
        movb    $122, -5(%ebp)
        movsbl  -5(%ebp), %eax
        pushl   %eax
        pushl   -4(%ebp)
        call    foo
        addl    $8, %esp
        movl    %eax, -12(%ebp)
        movl    -12(%ebp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```asm
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```asm
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```asm
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```asm
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

CF	
PF	
ZF	
SF	
OF	

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

CF	
PF	
ZF	
SF	
OF	

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```asm
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```

C Function calls in Assembly (64-bit)

```c
int foo(int a, char b) {
    if (b == 'z')
    {
        return a;
    }
    return 0;
}

int main(void)
{
    int a = 100;
    char b = 'z';
    int res = foo(a,b);
    return res;
}
```

```
foo:
        pushq   %rbp
        movq    %rsp, %rbp
        movl    %edi, -4(%rbp)
        movl    %esi, %eax
        movb    %al, -8(%rbp)
        cmpb    $122, -8(%rbp)
        jne     .L2
        movl    -4(%rbp), %eax
        jmp     .L3
.L2:
        movl    $0, %eax
.L3:
        popq    %rbp
        ret
main:
        pushq   %rbp
        movq    %rsp, %rbp
        subq    $16, %rsp
        movl    $100, -4(%rbp)
        movb    $122, -5(%rbp)
        movsbl  -5(%rbp), %edx
        movl    -4(%rbp), %eax
        movl    %edx, %esi
        movl    %eax, %edi
        call    foo
        movl    %eax, -12(%rbp)
        movl    -12(%rbp), %eax
        leave
        ret
```