

X86-64 Calling Conventions

Unix (e.g. macOS and Linux)

Passing Parameters:

- First six integer/pointer parameters (in order): %rdi, %rsi, %rdx, %rcx, %r8, %r9
- Additional parameters are passed on the stack, pushed in right-to-left (reverse) order.

Return Value:

- %rax (for 64-bit result), %eax (for 32-bit result)

Caller-Saved Registers (may be overwritten by called function):

- %rax, %rcx, %rdx, %rdi, %rsi, %rsp, %r8, %r9, %r10, %r11

Callee-Saved Registers (must be preserved – or saved and restored – by called function):

- %rbx, %rbp, %r12, %r13, %r14, %r15

Microsoft (e.g. for Windows)

Passing Parameters:

- First four integer/pointer parameters (in order): %rcx, %rdx, %r8, %r9
- Additional parameters are passed on the stack, pushed in right-to-left (reverse) order.
- Important: The caller must allocate 32 bytes on stack (by subtracting 32 from %rsp) right before calling the function. Don't forget to restore the %rsp (by adding 32) after the call.

Return Value:

- %rax (for 64-bit result), %eax (for 32-bit result)

Caller-Saved Registers (may be overwritten by called function):

- %rax, %rcx, %rdx, %r8, %r9, %r10, %r11

Callee-Saved Registers (must be preserved – or saved and restored – by called function):

- %rbx, %rbp, %rdi, %rsi, %rsp, %r12, %r13, %r14, %r15