Assembly in macOS: Hello World

This NASM code creates a “Hello, World!”program. It uses system calls to write the message to the console and then exit. The `.text` section holds the executable instructions, starting at `_start`. It first sets up the `sys_write` call (printing to stdout) and then the `sys_exit` call (exiting cleanly). The `.data` section stores the message itself and calculates its length.

```
global _start
section .text

_start:
    mov     rax, 0x2000004  ; sys_write
    mov     rdi, 1          ; stdout
    lea     rsi, [rel msg]
    mov     rdx, msg.len
    syscall

    mov     rax, 0x2000001  ; sys_exit
    xor     rdi, rdi        ; exit code 0
    syscall

section .data
msg:    db  "Hello, World!", 10
.len:   equ $ - msg
```

```
% ld -o hello -e _start hello.o -lSystem -L$(xcrun --show-sdk-path)/usr/lib -platform_version macos 15.2 15.2
```

```
%clang -o hello hello.o -nostdlib -e _start -Wl,-platform_version,macos,10.15,10.15
```