Hướng dẫn cơ bản: Các lệnh Jump trong Assembly x86_64

Trong Assembly x86_64, các lệnh jump (nhảy) được dùng để điều khiển luồng chương trình. Chúng thường dựa vào các cờ (flags) trong thanh ghi RFLAGS:
- ZF (Zero Flag)
- CF (Carry Flag)
- SF (Sign Flag)
- OF (Overflow Flag)
- PF (Parity Flag)

Dưới đây là bảng tổng hợp các lệnh jump phổ biến, cờ liên quan, và ví dụ ngắn gọn.

Lệnh	Điều kiện	Cờ sử dụng	Ví dụ
jmp label	Nhảy vô điều kiện	-	jmp start
je / jz	Bằng (Equal / Zero)	ZF=1	cmp rax, rbx je equal_case
jne / jnz	Khác (Not equal)	ZF=0	cmp rax, rbx jne not_equal
ja / jnbe	Lớn hơn (unsigned)	CF=0 & ZF=0	cmp rax, rbx ja bigger
jae / jnb	≥ (unsigned)	CF=0	cmp rax, rbx jae bigger_eq
jb / jnae	Nhỏ hơn (unsigned)	CF=1	cmp rax, rbx jb smaller
jbe / jna	≤ (unsigned)	CF=1 or ZF=1	cmp rax, rbx jbe small_eq
jg / jnle	Lớn hơn (signed)	ZF=0 & SF=OF	cmp rax, rbx jg greater
jge / jnl	≥ (signed)	SF=OF	cmp rax, rbx jge greater_eq
jl / jnge	Nhỏ hơn (signed)	SF≠OF	cmp rax, rbx

			jl less
jle / jng	≤ (signed)	ZF=1 or SF≠OF	cmp rax, rbx jle less_eq
jc	Carry = 1	CF=1	jc carry_case
jnc	No Carry	CF=0	jnc no_carry
jo	Overflow	OF=1	jo overflow_case
jno	No Overflow	OF=0	jno no_overflow
js	Sign (âm)	SF=1	js negative_case
jns	No Sign (dương)	SF=0	jns positive_case
jp / jpe	Parity even	PF=1	jp even_case
jnp / jpo	Parity odd	PF=0	jnp odd_case
jecxz / jrcxz	ECX/RCX=0	ECX/RCX	jecxz loop_end
loop	RCX--, nếu RCX!=0 thì nhảy	RCX	mov rcx,10 loop label

Kết luận:
- Dùng ZF cho so sánh bằng/khác.
- Dùng CF cho so sánh không dấu (unsigned).
- Dùng SF & OF cho so sánh có dấu (signed).
- Các lệnh như loop, jecxz giúp viết vòng lặp dễ dàng.