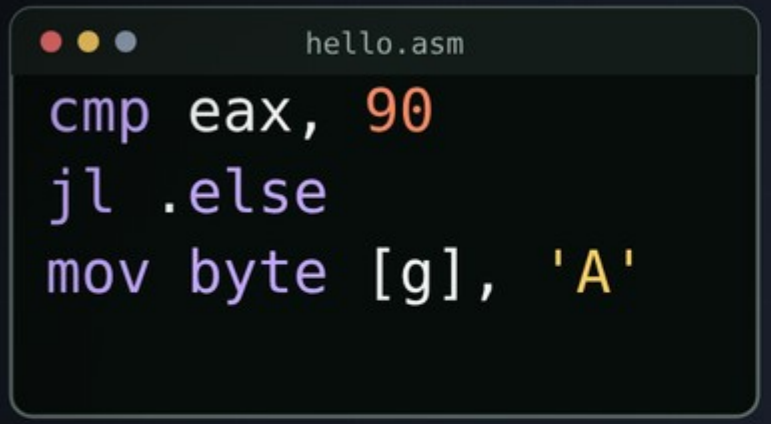
LESSON 6



ASM

cmp · jmp · loop

Learn it all @MyWebUniversity



A FREE MYWEBUNIVERSITY COURSE

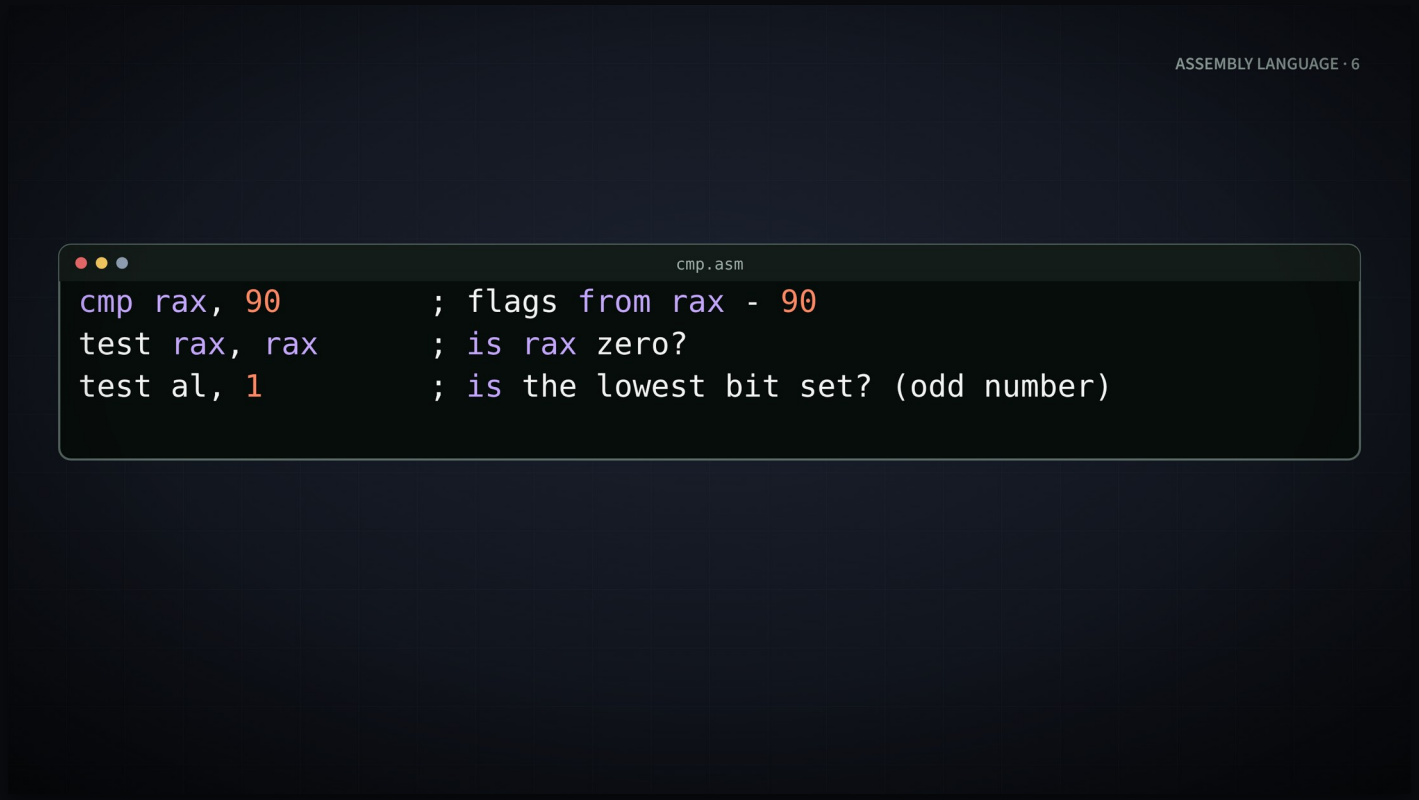
ASSEMBLY LANGUAGE

LESSON 6

Comparisons, Jumps and Loops

cmp, test, jmp, conditional jumps, if/else and loops in assembly

cmp and test



- cmp subtracts the second operand from the first, sets the flags, and throws the result …
- test rax, rax is the standard way to check whether a register is zero.

01

Conditional Jumps

Signed	Unsigned	Jump if
je / jz	je / jz	equal / zero
jne	jne	not equal
jl	jb	less / below
jg	ja	greater / above
jle · jge	jbe · jae	≤ · ≥

- jmp always jumps to a label. Conditional jumps jump only when the flags match: je if equal, ...
- Use jl, jg, jle and jge for signed numbers, and jb, ja, jbe and jae for unsigned numbers.

02

if / else in Assembly

grade.asm

```
; if (score >= 90) grade = 'A'; else grade = 'B';
    mov eax, [score]
    cmp eax, 90
    jl  .else               ; opposite condition
    mov byte [grade], 'A'
    jmp .done
.else:
    mov byte [grade], 'B'
.done:
```

- An if becomes a compare and a jump around the code you want to skip. Usually you jump on …
- Here is a grade check: if the score is at least ninety, store A, otherwise store B.

03

Loops

sum.asm

```
; sum = 0; for (i = 0; i < 10; i++) sum += nums[i];
    xor eax, eax            ; sum = 0
    xor ecx, ecx            ; i = 0
.loop:
    add eax, [nums + rcx*4]
    inc ecx
    cmp ecx, 10
    jl  .loop
    mov [sum], eax
```

- A loop is a label, a body, a counter update, and a conditional jump back. Here we add up an …
- Labels starting with a dot are local to the nearest normal label, so you can reuse names …

04

Branchless Code with cmov

ASSEMBLY LANGUAGE · 6

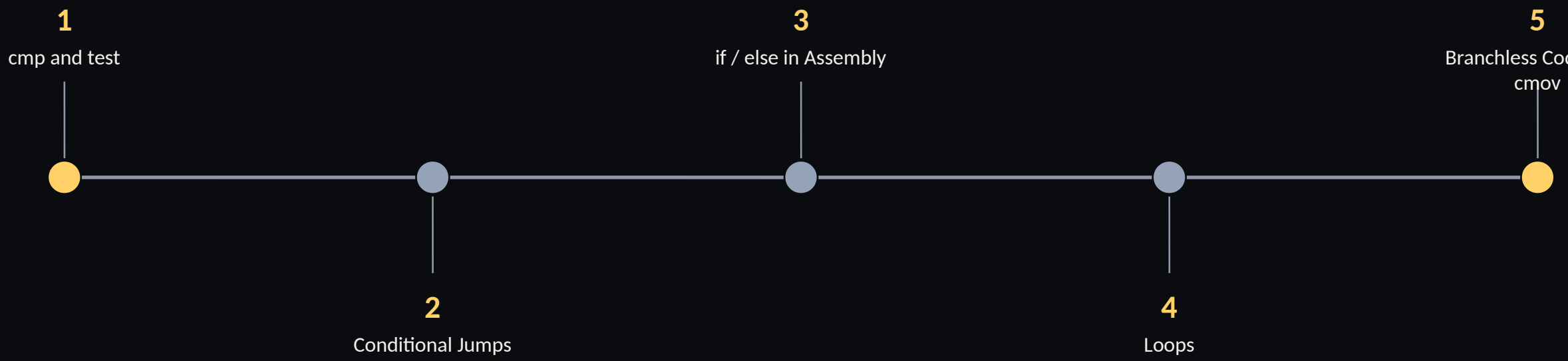
max.asm

```
; rax = max(rdi, rsi)
    mov rax, rdi
    cmp rsi, rdi
    cmovg rax, rsi          ; if rsi > rdi, rax = rsi
    ret
```

- Modern CPUs guess which way a jump goes. A wrong guess costs time. cmov moves a value only …
- Compilers use it for simple choices like max and min.

05

Lesson Map


1
cmp and test
2
Conditional Jumps
3
if / else in Assembly
4
Loops
5
Branchless Code with cmov

Sources and Further Reading

ASSEMBLY LANGUAGE · 6

cmp.asm

```
cmp rax, 90         ; flags from rax - 90
test rax, rax       ; is rax zero?
test al, 1          ; is the lowest bit set? (odd number)
```

ASSEMBLY LANGUAGE · 6

grade.asm

```
; if (score >= 90) grade = 'A'; else grade = 'B';
    mov eax, [score]
    cmp eax, 90
    jl  .else               ; opposite condition
    mov byte [grade], 'A'
    jmp .done
.else:
    mov byte [grade], 'B'
.done:
```

Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari