Princeton University
COS 217: Introduction to Programming Systems
IA-32 Condition Codes

Condition Codes

Bits in the EFLAGS register

`cmpl` *`src, dest`*

> Performs the subtraction *dest* - *src*, and sets the condition codes depending upon the difference:

Condition Code	Set When
ZF (zero flag)	The difference is 0
SF (sign flag)	The difference is negative, that is, the high order bit of the difference is 1
CF (carry flag)	The difference is mathematically incorrect when we view the operands as **unsigned** integers
OF (overflow flag)	The difference is mathematically incorrect when we view the operands as **signed** integers

Conditional Control Transfer Instructions (Used After Comparing Unsigned Numbers)

Instruction	Jump if and only if
`je  (jump iff equal)`	`ZF`
`jne (jump iff not equal)`	`~ZF`
`jb  (jump iff below)`	`CF`
`jae (jump iff above or equal)`	`~CF`
`jbe (jump iff below or equal)`	`CF | ZF`
`ja  (jump iff above)`	`~(CF | ZF)`

Examples (assuming a 5-bit computer for simplicity):

Comparison	Subtraction	Resulting Condition Codes	Execution of jb
12 and 6	01100 12 -00110 -6 ----- -- 00110 6	CF = 0 (unsigned diff is correct)	CF == 0 So don't jump
6 and 12	00110 6 -01100 -12 ----- -- 11010 26	CF = 1 (unsigned diff is incorrect)	CF == 1 So jump

Conditional Control Transfer Instructions (Used After Comparing Signed Numbers)

Instruction	Jump if and only if
je (jump iff equal)	ZF
jne (jump iff not equal)	~ZF
jl (jump iff less than)	SF ^ OF
jge (jump iff greater than or equal)	~(SF ^ OF)
jle (jump iff less than or equal)	(SF ^ OF) | ZF
jg (jump iff greater than)	~((SF ^ OF) | ZF)

Examples (assuming a 5-bit computer for simplicity):

Comparison	Subtraction	Resulting Condition Codes	Execution of jl
12 and 6	01100 12 -00110 -6 ----- -- 00110 6	SF = 0 (diff high order bit is 0) OF = 0 (signed diff is correct)	(SF ^ OF) == 0 So don't jump
-6 and -12	11010 -6 -10100 --12 ----- -- 00110 6	SF = 0 (diff high order bit is 0) OF = 0 (signed diff is correct)	(SF ^ OF) == 0 So don't jump
6 and 12	00110 6 -01100 -12 ----- -- 11010 -6	SF = 1 (diff high order bit is 1) OF = 0 (signed diff is correct)	(SF ^ OF) == 1 So jump
-12 and 6	10100 -12 -11010 --6 ----- -- 11010 -6	SF = 1 (diff high order bit is 1) OF = 0 (signed diff is correct)	(SF ^ OF) == 1 So jump
-12 and 6	10100 -12 -00110 -6 ----- -- 01110 14	SF = 0 (diff high order bit is 0) OF = 1 (signed diff is incorrect)	(SF ^ OF) == 1 So jump
-6 and 12	11010 -6 -01100 -12 ----- -- 01110 14	SF = 0 (diff high order bit is 0) OF = 1 (signed diff is incorrect)	(SF ^ OF) == 1 So jump
6 and -12	00110 6 -10100 --12 ----- -- 10010 -14	SF = 1 (diff high order bit is 1) OF = 1 (signed diff is incorrect)	(SF ^ OF) == 0 So don't jump
12 and -6	01100 12 -11010 --6 ----- -- 10010 -14	SF = 1 (diff high order bit is 1) OF = 1 (signed diff is incorrect)	(SF ^ OF) == 0 So don't jump