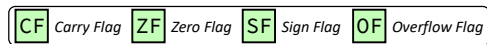


Implicit Setting (Review)

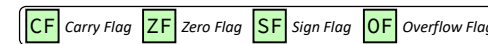
- ❖ *Implicitly* set by **arithmetic** and **logical** operations as side effect
 - Notable exception: `leaq` (beware! 😊)
- ❖ Example: `%al=0x80`, execute **`addb %al, %al`**



11

Explicit Setting (Review)

- ❖ *Explicitly* set by **`cmp`** and **`test`**, whose purposes are to change the condition codes *without storing their results*
 - `cmp` is equivalent result to `sub (-)`; `test` is equivalent result to `and (&)`
- ❖ Example: `%al=0x83` and `%bl=0x8C`, execute **`testb %al, %bl`**



12

Choosing Instruction Pairs

- ❖ Find correct form of condition in table, use corresponding instrs
 - Instruction #1 = chosen operation, Instruction #2 = row heading
 - Recall: `cmp` performs `sub`
`test` performs `and`

Examples:

- `x >= y` → `x - y >= 0`
→ **`cmp y, x`**
`jge target`
- `x == 3` → `x - 3 == 0`
`cmp 3, x`
`jle target`
- `x & x != 0` → **`test x, x`**
`jne target`

		(op) s, d
<code>je /sete</code>	"Equal"	d (op) s == 0
<code>jne/setne</code>	"Not equal"	d (op) s != 0
<code>js /sets</code>	"Signed" (negative)	d (op) s < 0
<code>jns/setns</code>	"Not signed" (nonnegative)	d (op) s >= 0
<code>jb /setg</code>	"Greater"	d (op) s > 0
<code>jge/setge</code>	"Greater or equal"	d (op) s >= 0
<code>jl /setl</code>	"Less"	d (op) s < 0
<code>jle/setle</code>	"Less or equal"	d (op) s <= 0
<code>ja /seta</code>	"Above" (unsigned >)	d (op) s > 0U
<code>jb /setb</code>	"Below" (unsigned <)	d (op) s < 0U

16

Polling Question

Register	Use(s)
<code>%rdi</code>	1 st argument (x)
<code>%rsi</code>	2 nd argument (y)
<code>%rax</code>	return value

- A. **`cmpq %rsi, %rdi`**
`jle .L4`
- B. **`cmpq %rsi, %rdi`**
`jb .L4`
- C. **`testq %rsi, %rdi`**
`jle .L4`
- D. **`testq %rsi, %rdi`**
`jb .L4`

```
long absdiff(long x, long y) {
    long result;
    if (x > y)
        result = x-y;
    else
        result = y-x;
    return result;
}
```

```
absdiff:
    _____
    _____ # x > y:
    movq    %rdi, %rax
    subq    %rsi, %rax
    ret
.L4:
    movq    %rsi, %rax
    subq    %rdi, %rax
    ret
```

21