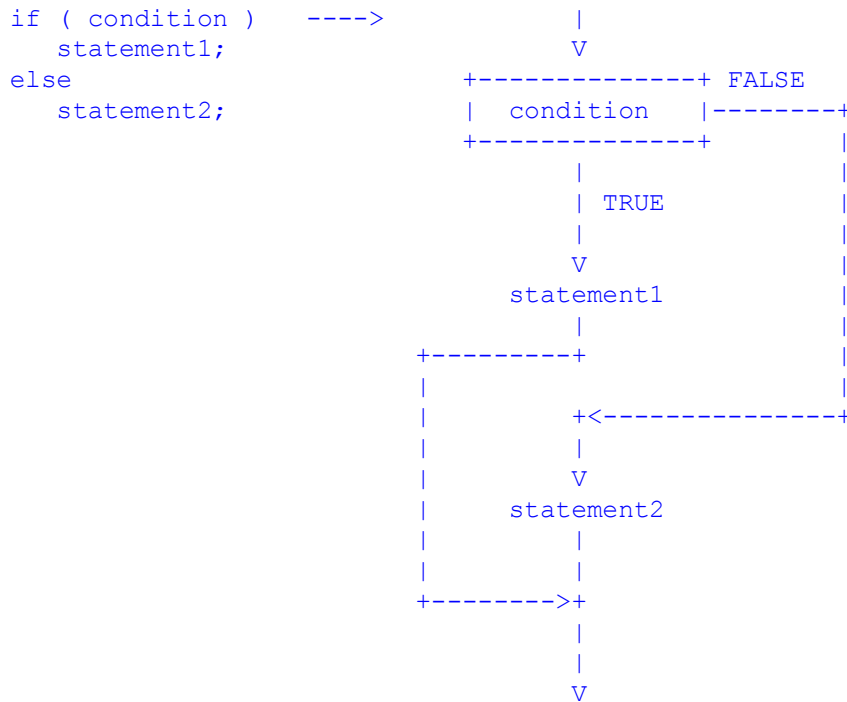
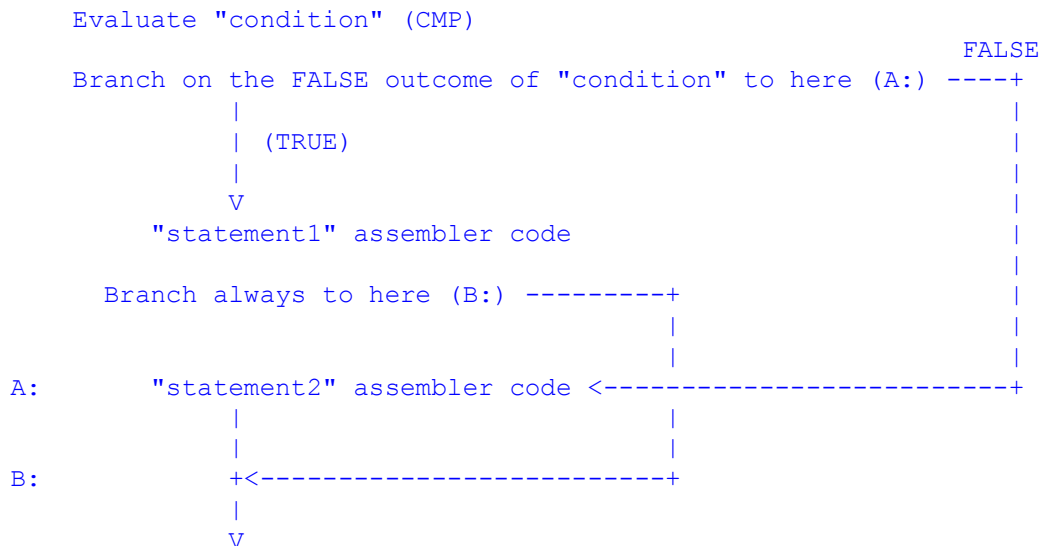

The simple if-else statement

- **Program control** of the **if-else** statement:



- Assembler construct that realizes the control flow of the **if-else** statement is:



- **Example: find the maximum of 2 integers**

int x;	Assembler construct for this if-statement:
int y;	
int max;	
if (x >= y)	MOVE.L x, D0
max = x;	CMP.L y, D0 Compares x against y
else	BLT L1 Skip when x < y
max = y;	

```

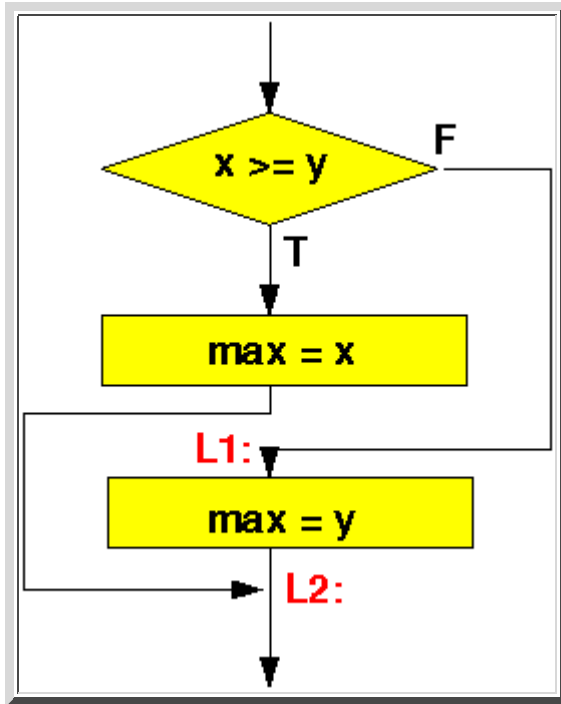
MOVE.L x, max
BRA    L2      Must skip over else part !

L1: MOVE.L y, max

L2:

```

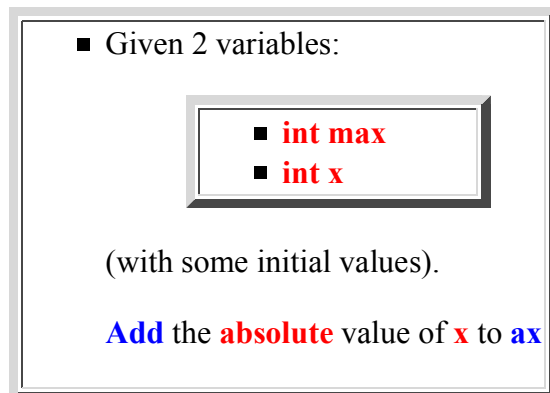
The **flow chart** of the **above program** is:



- Example assembler program: [click here](#)

- **Another example**

- Program:



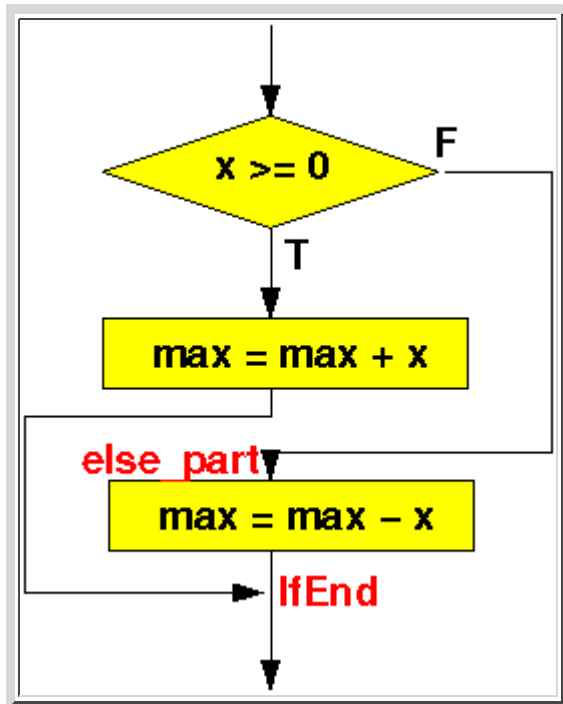
- Program in **Java**:

```

if ( x >= 0 )
    max = max + x;
else
    max = max - x;

```

The **flow chart** of the above program is:



○ Program in **M68000 assembler code**:

```

    move.l   x,d0
    cmp.l   #0,d0          * Computes: d0 - 0 = x - 0

    blt     else_part     * if ( x - 0 < 0 ) branch to else_part

* ----- Continue if blt fails to branch
* This happens if x-0 >=0, or: x >= 0

then_part:
    move.l   max, d1
    add.l   d0, d1
    move.l   d1, max

    bra     IfEnd         branch over ElsePart to continue at IfEnd
*                               (Otherwise we execute the ELSE part !!!)

* ----- Arrives here if we branched
* This happens when x-0 < 0, or: x < 0
else_part:
    move.l   max, d1
    sub.l   d0, d1
    move.l   d1, max

IfEnd:

```

(program continues here if there are more statements...)

- **Example Program:** (Demo above code)

Example

- Prog file: [click here](#)

How to run the program:

- **Right click** on link and **save** in a scratch directory
 - To compile: `as255 if-else2`
 - To run: use `m68000`
-
-