



Comparaciones en lenguaje ensamblador

Objetivo: Entender cómo se realizan las comparaciones de datos en lenguaje ensamblador

En lenguaje ensamblador no existe la instrucción IF como se trabaja en otros lenguajes para comparar y tomar decisiones, para esto se utiliza `cmp` en conjunto con los saltos.

cmp: esta instrucción se usa para comparar pero solo hace la comparación, no realiza ninguna toma de decisiones para ello tiene que hacer uso de los saltos.

¿Qué es un salto? Es una instrucción de lenguaje ensamblador que envía el control del programa a un conjunto de instrucciones que se identifican por medio de una etiqueta

Ejemplo: Si se quiere saber si un número es mayor de 5

Vamos a enviar el número 10 al registro a la zona baja del registro AX es decir a AL y luego lo vamos a comparar con otro número por ejemplo el 5

```
mov AL, 10 ; movemos el número 10 a AL
```

```
cmp AL, 5; comparamos si el número que está almacenado en AL es mayor a 5
```

```
jg mayor; Si AL es mayor que 5 salta a la etiqueta que se llama mayor, es un salto condicional
```

```
jmp menor; En caso contrario salta a la etiqueta menor. Es decir si AL no es mayor que 5 salta a la etiqueta menor. Jmp es un salto incondicional, lo que significa que no necesita de una comparación previa para saltar. Salta sin que exista una comparación previa o una condición.
```

Por otro lado el salto se hace a una etiqueta que va con dos puntos (:). Y debajo de la etiqueta se coloca el código. Para este ejemplo se han establecido dos etiquetas una etiqueta denominada **mayor** para ubicar el código que se va a ejecutar cuando la condición sea verdadera y otra etiqueta **menor** para ubicar el código que se va a ejecutar cuando la condición sea falsa.

Las etiquetas son: Nombres que se les asignan a las direcciones de cualquier instrucción o localización de memoria. Una etiqueta toma el lugar de una dirección. Tan pronto como el ensamblador encuentra una etiqueta, esta es remplazada por la dirección correcta.

Entonces el código iría quedando así:

```
mov AL, 10
```

```
cmp AL, 5
```

```
jg mayor
```

```
jmp menor
```

```
mayor:
```



aquí en esta sección va el código que es ejecutara si es mayor a 2
menor:

aquí en esta sección va el código que es ejecutara si es menor

Entonces, la “cmp” es quien compara y quien toma la decisión es el **salto**.

Algunos tipos de saltos:

JG salta si es mayor

JGE salta si es mayor o igual

JZ salta si es cero

JE Salta si es igual

JL Salta si es menor

JLE Salta si es menor o igual

JNZ Salta si el valor calculado no es cero

JS Salta si el valor calculado es negativo

JNS Salta si el valor calculado no es negativo

Estos son algunos saltos que después de que se hace la comparación (cmp) se colocan para que tomen la decisión de ir a una ***Etiqueta con base al resultado de una comparación.***

En caso que se necesite un salto incondicional se usa ***jmp*** que salta sin necesidad de evaluar una condición.

Instrucciones tipo Salto



Instrucción			Condición
JA	Jump if Above	salta si por encima	$A > B$ (sin signo)
JAE	Jump if Above or Equal	salta si por encima o igual	$A \geq B$ (sin signo)
JB	Jump if Below	salta si por debajo	$A < B$ (sin signo)
JBE	Jump if Below or Equal	salta si por debajo o igual	$A \leq B$ (sin signo)
JE	Jump if Equal	salta si igual	$A = B$
JG	Jump if Greater	salta si mayor	$A > B$ (con signo)
JGE	Jump if Greater or Equal	salta si mayor o igual	$A \geq B$ (con signo)
JL	Jump if Less	salta si menor	$A < B$ (con signo)
JLE	Jump if Less or Equal	salta si menor o igual	$A \leq B$ (con signo)

INSTRUCCIONES TIPO SALTO																																																											
¿Para qué son?:	Producen un cambio en la ejecución del programa pasando a una nueva posición.																																																										
Instrucción:	La orden es JMP (de Jump, salto)																																																										
Formato general:	[etiqueta:] JMP dirección corta, cercana o lejana																																																										
Salto incondicional	Salto condicional																																																										
<p>Salta sin comprobar ninguna condición.</p> <p>Los nemotécnicos más frecuentes: <i>jump</i> o <i>branch</i>.</p> <p>Hay más de 30 instrucciones de salto; sin embargo, las etiquetas más comunes son:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #c6e0b4;"> <th>Instrucción</th> <th>Descripción</th> </tr> </thead> <tbody> <tr><td>JMP</td><td>Salto incondicional directo</td></tr> <tr><td>JE</td><td>Salto por igual</td></tr> <tr><td>JNE</td><td>Salto por no igual</td></tr> <tr><td>JG</td><td>Salto por mayor</td></tr> <tr><td>JNG</td><td>Salto por no mayor</td></tr> <tr><td>JGE</td><td>Salto por mayor igual</td></tr> <tr><td>JNGE</td><td>Salto por no mayor igual</td></tr> <tr><td>JL</td><td>Salto por menor</td></tr> <tr><td>JNL</td><td>Salto por no menor</td></tr> <tr><td>JLE</td><td>Salto por menor igual</td></tr> <tr><td>JNLE</td><td>Salto por no menor igual</td></tr> <tr><td>JS</td><td>Salto negativo</td></tr> </tbody> </table>	Instrucción	Descripción	JMP	Salto incondicional directo	JE	Salto por igual	JNE	Salto por no igual	JG	Salto por mayor	JNG	Salto por no mayor	JGE	Salto por mayor igual	JNGE	Salto por no mayor igual	JL	Salto por menor	JNL	Salto por no menor	JLE	Salto por menor igual	JNLE	Salto por no menor igual	JS	Salto negativo	<p>Salta si la condición se cumple.</p> <p>Los nemotécnicos más frecuentes: <i>jcond</i> o <i>bcond</i></p> <p>Algunas etiquetas para instrucciones de salto condicional son:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #c6e0b4;"> <th>Instrucción</th> <th>Descripción</th> </tr> </thead> <tbody> <tr><td>JO</td><td>Jump if overflow.</td></tr> <tr><td>JNO</td><td>Jump if not overflow.</td></tr> <tr><td>JC, JNAE, JB</td><td>Jump if Carry, Jump if Not Above or Equal y Jump if Below.</td></tr> <tr><td>JNC, JAE, JNB</td><td>Jump if not Carry, Jump if Above or Equal y Jump if Not Below.</td></tr> <tr><td>JNZ o JNE</td><td>Jump if Not Zero o Jump if Not Equal.</td></tr> <tr><td>JBE o JNA</td><td>Jump if Below or Equal o Jump if Not Above.</td></tr> <tr><td>JA o JNBE</td><td>Jump if Above o Jump if Not Below of Equal.</td></tr> <tr><td>JS</td><td>Jump if Sign.</td></tr> <tr><td>JNS</td><td>Jump if Not Sign.</td></tr> <tr><td>JP, JPE</td><td>Jump if Parity o Jump if Parity Even.</td></tr> <tr><td>JNP, JPO</td><td>Jump if Not Parity, Jump if Parity Odd.</td></tr> <tr><td>JL, JNGE</td><td>Jump if Less, Jump if Not Greater of Equal.</td></tr> <tr><td>JGE, JNL</td><td>Jump if Greater or Equal, Jump if Not Less.</td></tr> <tr><td>JLE, JNG</td><td>Jump if Lower or Equal, Jump if Not Greater.</td></tr> <tr><td>JG, JNLE</td><td>Jump if Greater, Jump if Not Lower or Equal.</td></tr> </tbody> </table>	Instrucción	Descripción	JO	Jump if overflow.	JNO	Jump if not overflow.	JC, JNAE, JB	Jump if Carry, Jump if Not Above or Equal y Jump if Below.	JNC, JAE, JNB	Jump if not Carry, Jump if Above or Equal y Jump if Not Below.	JNZ o JNE	Jump if Not Zero o Jump if Not Equal.	JBE o JNA	Jump if Below or Equal o Jump if Not Above.	JA o JNBE	Jump if Above o Jump if Not Below of Equal.	JS	Jump if Sign.	JNS	Jump if Not Sign.	JP, JPE	Jump if Parity o Jump if Parity Even.	JNP, JPO	Jump if Not Parity, Jump if Parity Odd.	JL, JNGE	Jump if Less, Jump if Not Greater of Equal.	JGE, JNL	Jump if Greater or Equal, Jump if Not Less.	JLE, JNG	Jump if Lower or Equal, Jump if Not Greater.	JG, JNLE	Jump if Greater, Jump if Not Lower or Equal.
Instrucción	Descripción																																																										
JMP	Salto incondicional directo																																																										
JE	Salto por igual																																																										
JNE	Salto por no igual																																																										
JG	Salto por mayor																																																										
JNG	Salto por no mayor																																																										
JGE	Salto por mayor igual																																																										
JNGE	Salto por no mayor igual																																																										
JL	Salto por menor																																																										
JNL	Salto por no menor																																																										
JLE	Salto por menor igual																																																										
JNLE	Salto por no menor igual																																																										
JS	Salto negativo																																																										
Instrucción	Descripción																																																										
JO	Jump if overflow.																																																										
JNO	Jump if not overflow.																																																										
JC, JNAE, JB	Jump if Carry, Jump if Not Above or Equal y Jump if Below.																																																										
JNC, JAE, JNB	Jump if not Carry, Jump if Above or Equal y Jump if Not Below.																																																										
JNZ o JNE	Jump if Not Zero o Jump if Not Equal.																																																										
JBE o JNA	Jump if Below or Equal o Jump if Not Above.																																																										
JA o JNBE	Jump if Above o Jump if Not Below of Equal.																																																										
JS	Jump if Sign.																																																										
JNS	Jump if Not Sign.																																																										
JP, JPE	Jump if Parity o Jump if Parity Even.																																																										
JNP, JPO	Jump if Not Parity, Jump if Parity Odd.																																																										
JL, JNGE	Jump if Less, Jump if Not Greater of Equal.																																																										
JGE, JNL	Jump if Greater or Equal, Jump if Not Less.																																																										
JLE, JNG	Jump if Lower or Equal, Jump if Not Greater.																																																										
JG, JNLE	Jump if Greater, Jump if Not Lower or Equal.																																																										
<div style="display: flex; flex-direction: column; align-items: flex-start;"> <div style="background-color: #e67e22; padding: 5px; margin-bottom: 5px; border-radius: 5px;"> <p>Salto corto</p> <ul style="list-style-type: none"> Una operación JMP a una etiqueta dentro de -128 a +127 bytes es un salto corto. </div> <div style="background-color: #f1c40f; padding: 5px; margin-bottom: 5px; border-radius: 5px;"> <p>Salto cercano</p> <ul style="list-style-type: none"> Una operación JMP que excede -128 a +127 bytes se convierte en un salto cercano. </div> <div style="background-color: #27ae60; padding: 5px; border-radius: 5px;"> <p>Salto lejano</p> <ul style="list-style-type: none"> Si el destino es un procedimiento con el atributo FAR, se convierte en un salto lejano. </div> </div> <div style="margin-top: 20px; text-align: center;"> <div style="background-color: #f39c12; padding: 10px; border-radius: 10px; display: inline-block;">El ensamblador genera la longitud de cada instrucción</div> <div style="background-color: #6b4594; color: white; padding: 10px; border-radius: 10px; display: inline-block; margin-left: 20px;">Cada instrucción JMP puede ser de dos o tres bytes</div> </div>																																																											
<div style="display: flex; align-items: center;"> <div style="border: 1px solid #34495e; padding: 5px; margin-right: 10px;">Instrucciones TEST</div> <div style="display: flex; flex-direction: column; gap: 5px;"> <div style="background-color: #3498db; padding: 5px; border-radius: 5px;">Propósito</div> <div style="background-color: #3498db; padding: 5px; border-radius: 5px;">Sintaxis</div> <div style="background-color: #3498db; padding: 5px; border-radius: 5px;">Descripción</div> </div> </div> <div style="margin-top: 10px;"> <p style="text-align: center;">Comparar lógicamente los operandos</p> <p style="text-align: center;">TEST destino, fuente</p> <p>Realiza una conjunción, bit por bit, de los operandos, pero a diferencia de AND esta instrucción no coloca el resultado en el operando destino, solo tiene efecto sobre el estado de las banderas.</p> </div>																																																											
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="background-color: #34495e; color: white; padding: 10px; border-radius: 10px; font-weight: bold;">Bucles</div> <div style="background-color: #95a5a6; padding: 10px; border-radius: 10px; font-weight: bold;">La instrucción es LOOP.</div> <div style="background-color: #95a5a6; padding: 10px; border-radius: 10px; font-weight: bold;">Compara CX con cero.</div> <div style="background-color: #95a5a6; padding: 10px; border-radius: 10px; font-weight: bold;">Si es igual, sigue adelante.</div> <div style="background-color: #95a5a6; padding: 10px; border-radius: 10px; font-weight: bold;">Si no lo es, vuelve al lugar que se indica en su operando, decrementando CX en uno</div> </div>																																																											



Ciclo Loop (bucles).

Realiza n iteraciones mientras el registro CX sea mayor que cero.

Su procedimiento es el siguiente:

mov CX, tamaño; se establece el tamaño del ciclo, es decir el número de iteraciones a realizar

etiqueta: ; la etiqueta que denota el inicio del cuerpo del bucle

.....

 Código ensamblador; código ensamblador, parte del cuerpo del ciclo a iterar

.....

loop etiqueta ; Instrucción loop que retorna hacia la etiqueta que marca el inicio del cuerpo del ciclo mientras que $CX > 0$. En cada iteración CX decrementa en 1.