**Computer Organization & Assembly Language**

**Lab 06**

**Topics:**

1. Compare instruction
2. Conditional Jumps
3. While Loop
4. Conditional Loops

**Compare Instruction**

* Compare the values of two operand
* Syntax: CMP operand1, operand2
* Example

*Mov ax, 1000*

*Mov cx, 1000*

*Cmp cx, ax*

**Conditional Jumps**

* Conditional jump transfer control to another instruction when the condition is satisfied.
* Syntax: Jcond destination
* **Equality Comparison**

1. Jumps to label if equal: JE (left operand = Right Operand)
2. Jump if not equal: JNE (left operand != Right Operand)
3. Jump if zero: JZ
4. Jump if not zero: JNZ
5. Jump if cx zero: JCXZ

**Example**

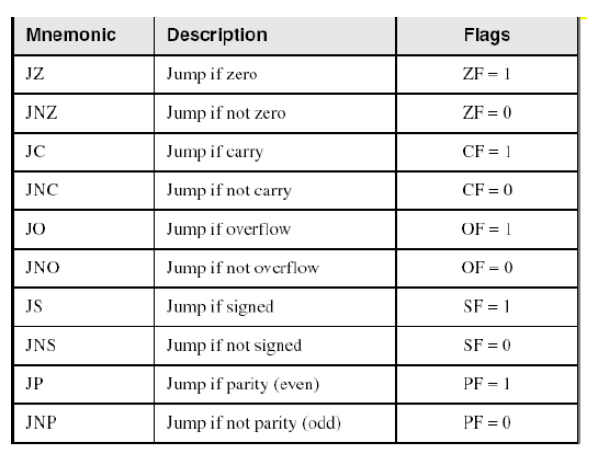
*Mov dx, 0A523h*

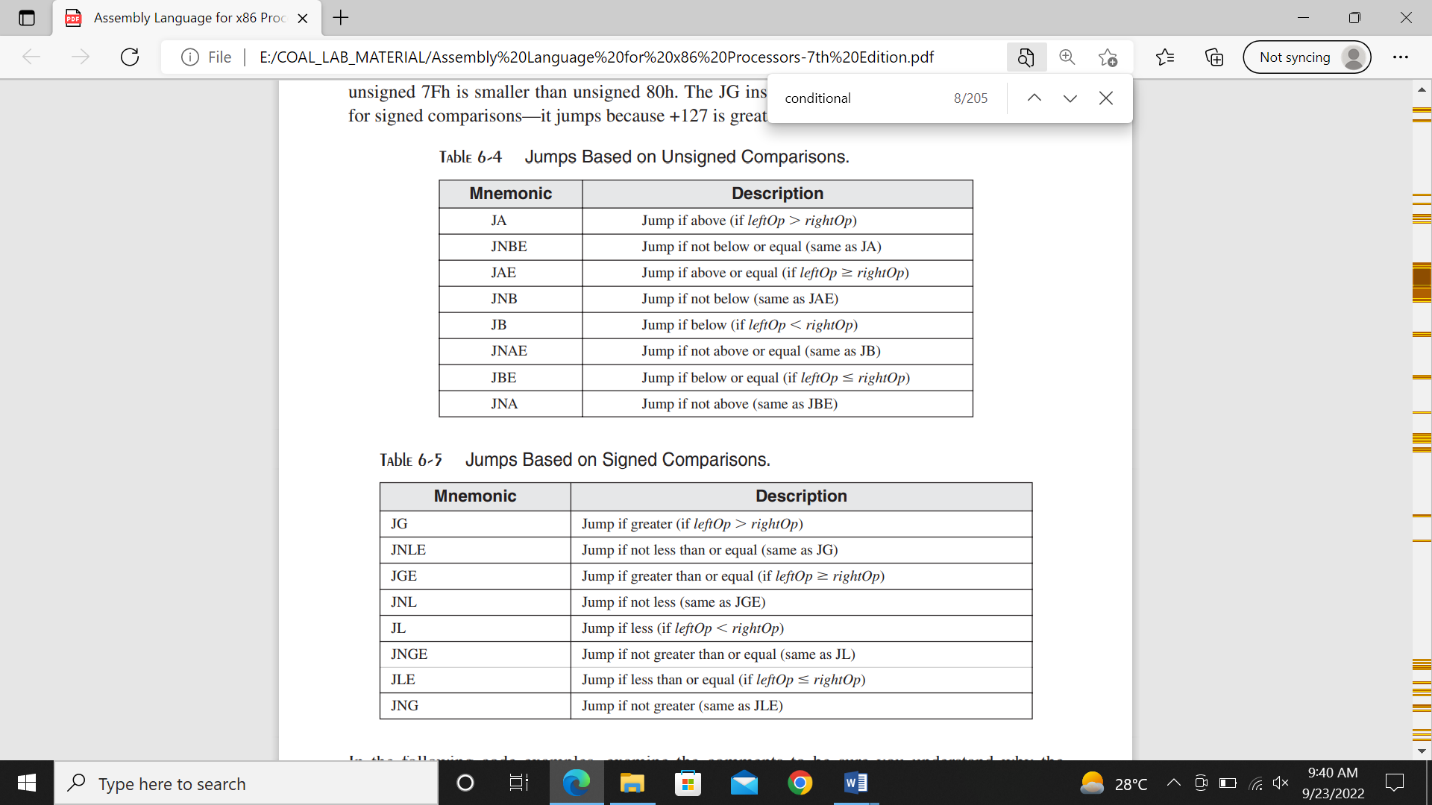
*Cmp dx, 0A523h*

*Jne l5*

*Je l1*

* **Jump Based on Specific Flag values**

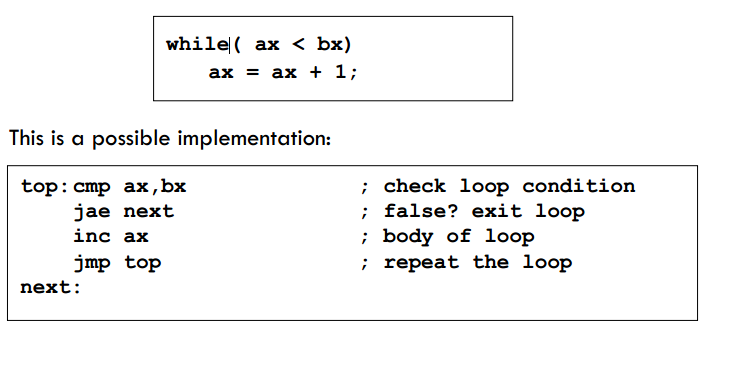
****

* **Jump Based on Unsigned Comparison**

**While Loop**

While loop tests a condition before performing a block of statements. As long as the condition remains true, statements are repeated.

**Example:**

****

1. **LOOPZ (Loop if Zero):**

The LOOPZ instruction is an x86 assembly language command that facilitates conditional looping in a program. It repeatedly executes a specific block of code as long as the zero flag (ZF) in the CPU's flags register is set. This instruction is particularly useful for scenarios where you want to iterate through a loop until a specific condition is met, and that condition involves checking if a certain result or variable is zero.

1. **LOOPE (Loop if Equal):**

LOOPE is another x86 assembly instruction that enables conditional looping. It continues to execute a designated code block as long as both the zero flag (ZF) is set and the counter register (typically CX or ECX) is not zero. LOOPE is often employed in scenarios where you need to repeatedly perform an action until a specific equality condition is satisfied, such as searching for a particular element in an array or list.

1. **LOOPNZ (Loop if Not Zero):**

The LOOPNZ instruction, similar to LOOPZ, is used for implementing conditional loops in assembly language programs. It repeatedly executes a specified block of code as long as the zero flag (ZF) is clear and the counter register (CX or ECX) is not zero. LOOPNZ is commonly employed when you want to iterate through a loop until a condition that involves checking if a value is not zero is met.

1. **LOOPNE (Loop if Not Equal):**

LOOPNE is an x86 assembly instruction that creates a conditional loop. It continues to execute a designated code block as long as both the zero flag (ZF) is clear and the counter register (typically CX or ECX) is not zero. LOOPNE is often used when you need to repeatedly perform an action until a specific inequality condition is satisfied, such as searching for a specific element in a data structure while ensuring it is not equal to the desired value.