**Flow Control Instructions and Branching Structures**

### **📘 Overview**

In here we learned how to **control the execution flow of an assembly program** using:

* **Unconditional jumps (JMP)**
* **Conditional jumps (JE, JG, JL, JGE, etc.)**
* The CMP instruction
* Branching logic structures like:  
  + IF-THEN
  + IF-THEN-ELSE
  + CASE (multi-way selection)

These are used to replicate **decision-making**, **looping**, and **structured programming logic** found in high-level languages like C, Python, or Java.

## **🔧 Detailed Concepts with Examples**

### **1️⃣ Unconditional Jump – JMP**

🔹 **What it is**:  
 Transfers control to another part of the code **regardless of any condition**.

🔹 **Why it's useful**:  
 Used to skip unnecessary code or to structure execution logically.

🔹 **Example:**

JMP SkipSection ; Always jumps to label SkipSection

MOV AH, 4 ; This line is skipped

MOV DL, 6 ; This too is skipped

SkipSection:

MOV DL, 7 ; Execution resumes here

### **2️⃣ Comparison Instruction – CMP**

🔹 **What it is**:  
 Compares two operands by subtracting them: destination - source  
 It **doesn’t store** the result, but it **sets flags** (Zero, Sign, Carry, Overflow) used for conditional jumps.

🔹 **Example:**

MOV AX, 5

CMP AX, 4 ; AX - 4 → sets flags

### **3️⃣ Conditional Jumps – Based on CMP**

🔹 **What they are**:  
 These instructions check the flags set by CMP to determine if a jump should happen.

| **Instruction** | **Meaning** |
| --- | --- |
| JE or JZ | Jump if Equal (Zero flag set) |
| JG | Jump if Greater (signed) |
| JL | Jump if Less (signed) |
| JGE | Jump if Greater or Equal |
| JLE | Jump if Less or Equal |

🔹 **Example: IF AX > 4**

MOV AX, 5

CMP AX, 4 ; 5 - 4 → result > 0 → jump

JG DoSomething

DoSomething:

; code here runs if AX > 4

### **4️⃣ IF-THEN Structure**

🔹 **What it is**:  
 A basic decision: if the condition is **true**, do something. If not, skip it.

🔹 **Example: Replace AX with 5 if it’s negative**

MOV AX, -3

CMP AX, 0

JGE Skip ; If AX >= 0, skip replacement

MOV AX, 5 ; Replaces negative value

Skip:

### **5️⃣ IF-THEN-ELSE Structure**

🔹 **What it is**:  
 Execute one block if a condition is true, another if it’s false.

🔹 **Example: Compare AL and BL**

MOV AL, 3

MOV BL, 4

CMP AL, BL

JL AL\_Less

JMP AL\_GreaterOrEqual

AL\_Less:

; Executes if AL < BL

JMP End

AL\_GreaterOrEqual:

; Executes if AL >= BL

End:

### **6️⃣ CASE (Multiway Branching)**

🔹 **What it is**:  
 Like a switch-case in C or Python match-case — jump to a block based on the value of a variable.

🔹 **Example: Print vowel if AL is 'A', 'E', 'I', 'O', 'U'**

MOV AL, 'E'

CMP AL, 'A'

JE IsVowel

CMP AL, 'E'

JE IsVowel

; ... (repeat for other vowels)

JMP NotVowel

IsVowel:

; AL is a vowel

NotVowel:

; AL is a consonant

## **🎯 Final Takeaways**

| **Concept** | **What I Can Now Do** |
| --- | --- |
| JMP | Skip or redirect code flow |
| CMP + Flags | Perform comparisons without storing the result |
| Conditional Jumps | Make logical decisions (like if, else) |
| IF-THEN, IF-ELSE | Replace or compare values conditionally |
| CASE | Handle multiple value-based decisions |
| Flags | Understand Zero, Sign, Carry, Overflow for flow control |