**Stacks**

## 

### **📘 Overview**

Here we learned how to use the **stack** in 8086 assembly language. The stack is a **LIFO (Last-In-First-Out)** data structure used for:

* Temporary data storage
* Reversing data
* Swapping values
* Summation
* Procedure support
* Character/string operations like reversal and palindrome checks

## **🔧 Key Concepts & Examples**

### **1️⃣ Stack Basics: .STACK, PUSH, POP**

#### **🔍 What I Learned:**

* .STACK reserves memory for stack use.
* PUSH adds (pushes) data onto the stack.
* POP retrieves (pops) the most recently pushed data.

#### **💡 Example:**

MOV AX, 1234h

MOV BX, 5678h

PUSH AX ; AX pushed first

PUSH BX ; Then BX

POP AX ; AX gets BX's value

POP BX ; BX gets AX's value → swapped

### **2️⃣ Using Stack to Swap Two Numbers**

#### **🔍 What I Learned:**

* You can swap values **without a temporary variable** using the stack.

#### **💡 Example:**

PUSH AX

PUSH BX

POP AX ; AX ← BX

POP BX ; BX ← AX

### **3️⃣ Summing Values Using Stack**

#### **🔍 What I Learned:**

* You can push values onto the stack and then pop them to calculate a sum.

#### **💡 Example:**

MOV CX, 5

MOV SI, 0

SumPush:

MOV AX, SI

ADD AX, 1 ; Simulate numbers 1 to 5

PUSH AX

INC SI

LOOP SumPush

MOV CX, 5

MOV BX, 0

SumPop:

POP AX

ADD BX, AX ; BX accumulates the sum

LOOP SumPop

### **4️⃣ Reversing a String Using Stack**

#### **🔍 What I Learned:**

* You can reverse the order of a string by pushing each character onto the stack and then popping them back into another buffer.

#### **💡 Example:**

LEA SI, str

MOV CX, 5

PushChars:

MOV AL, [SI]

PUSH AX

INC SI

LOOP PushChars

LEA DI, rev

MOV CX, 5

PopChars:

POP AX

MOV [DI], AL

INC DI

LOOP PopChars

### **5️⃣ Palindrome Check Using Stack**

#### **🔍 What I Learned:**

* A palindrome is a word that reads the same backward as forward.
* Push the original string onto the stack, then compare it character by character with the original from the front.

#### **💡 Example:**

LEA SI, str

MOV CX, 5

PushLoop:

MOV AL, [SI]

PUSH AX

INC SI

LOOP PushLoop

LEA SI, str

MOV CX, 5

CheckLoop:

POP BX

MOV AL, [SI]

CMP AL, BL

JNE NotPal

INC SI

LOOP CheckLoop

### **6️⃣ Using the Stack in Procedures**

#### **🔍 What I Learned:**

* The stack can be used to save data across **procedure calls**.
* Though not shown in full in this lab, this is **foundational** for calling subroutines and saving return addresses.

## **✅ Summary of Stack Instructions**

| **Instruction** | **Description** |
| --- | --- |
| .STACK | Declares stack segment memory |
| PUSH | Saves data to top of stack (SP decreases) |
| POP | Retrieves data from top of stack (SP increases) |
| PUSHF | Push CPU flags onto stack |
| POPF | Pop CPU flags from stack |
| CALL/RET | Internally use stack to manage control flow |