

CS401P FINAL TERM PREPARATION

صرف ایک ویڈیو اور۔۔۔
CS401P کی تیاری مکمل FINALTERM

CS401P Final Term Preparation
Complete in one video



BY SIR SHARJEEL KHALID

Subscribe (vu updates tech)

<http://www.youtube.com/@VuUpdatetech>

LAB 1: Installation of NASM & DOSBOX

(10 MCQs with explanations)

MCQ 1 (Single Correct)

Which assembler is officially used in CS401P labs?

- A. MASM
- B. TASM
- C. NASM ✓
- D. GAS

Explanation:

CS401P uses NASM (Netwide Assembler) because it is free, lightweight, and compatible with DOS-based environments.

MCQ 2 (Single Correct)

Why is DOSBOX required for CS401P labs?

- A. To write assembly code
- B. To debug C programs
- C. **To emulate DOS for NASM execution ✓**
- D. To compile Java programs

Explanation:

Modern 32/64-bit operating systems do not support DOS directly, so **DOSBOX emulates a DOS environment** for NASM.

MCQ 3 (Single Correct)

Which operating systems do NOT support NASM directly?

- A. 16-bit DOS
- B. **32-bit and 64-bit Windows ✓**
- C. MS-DOS
- D. FreeDOS

Explanation:

32-bit and 64-bit Windows lack real-mode DOS support, which is required by NASM.

MCQ 4 (Single Correct)

DOSBOX is best described as a:

- A. Compiler
- B. Debugger
- C. **DOS operating system emulator ✓**
- D. Text editor

Explanation:

DOSBOX emulates the DOS operating system, allowing old DOS-based programs to run on modern systems.

MCQ 5 (Conceptual)

What is the main purpose of Lab 1 in CS401P?

- A. Learning loops
- B. Learning sorting
- C. **Setting up the assembly programming environment** ✓
- D. Learning BIOS interrupts

Explanation:

Lab 1 focuses entirely on **installation and configuration** of NASM, DOSBOX, and editors.

MCQ 6 (Conceptual)

Which tool is mainly used to write assembly language code?

- A. DOSBOX
- B. Debugger
- C. **Programmer's Notepad / Notepad** ✓
- D. BIOS

Explanation:

Editors like **Notepad or Programmer's Notepad** are used to write `.asm` source files.

MCQ 7 (Conceptual)

Which component provides compatibility between NASM and modern Windows?

- A. NASM itself
- B. Programmer's Notepad
- C. **DOSBOX** ✓
- D. BIOS

Explanation:

DOSBOX bridges the compatibility gap between NASM and modern operating systems.

MCQ 8 (Conceptual)

What type of programming paradigm is assembly language?

- A. Object-oriented
- B. Functional
- C. **Low-level programming** ✓
- D. Declarative

Explanation:

Assembly language works close to hardware and registers, making it a **low-level language**.

MCQ 9 (Conceptual)

Why is NASM preferred in CS401P labs?

- A. It supports graphics
- B. It is Windows-only
- C. **It is simple and educational** ✓
- D. It replaces C language

Explanation:

NASM is simple and ideal for learning **core assembly concepts**.

MCQ 10 (Conceptual)

Which activity is emphasized during CS401P labs?

- A. Writing theory
- B. Watching lectures
- C. **Practicing code and sharing output** ✓
- D. Writing reports

Explanation:

Students write code, execute it, and **share screen/output** during lab sessions.

🔍 LAB 2: Loop Implementation in Assembly Language

Focus: Registers, Labels, Jumps, Loop Logic, Memory Variable

MCQ 1 (Single Correct)

Why are loops implemented manually in assembly language?

- A. Assembly does not support variables
- B. Assembly has no conditional statements
- C. **Assembly language has no built-in loop structures** ✓
- D. Assembly cannot repeat instructions

Explanation:

Assembly language does not have `for` or `while` loops. Repetition is achieved using **labels and jump instructions**.

MCQ 2 (Single Correct)

Which register is used as a loop counter in Lab 2?

- A. AX
- B. BX
- C. **CX** ✓
- D. DX

Explanation:

The **CX register** is traditionally used as a counter in loops. In Lab 2, it starts from 10 and is decremented each iteration.

MCQ 3 (Single Correct)

What is the initial value of register BX in Lab 2?

- A. 0
- B. **5** ✓
- C. 10
- D. 1

Explanation:

BX is initialized with 5, and this value is added to AX during each loop iteration.

MCQ 4 (Single Correct)

Which instruction is responsible for repeating the loop?

- A. JMP
- B. JE
- C. LOOP
- D. JNZ ✓

Explanation:

JNZ (Jump if Not Zero) jumps back to the label as long as the counter register is not zero.

MCQ 5 (Conceptual)

What is the main role of register AX in Lab 2?

- A. Loop control
- B. Memory addressing
- C. Accumulating the final result ✓
- D. Holding loop count

Explanation:

AX starts from 0 and accumulates the sum by repeatedly adding the value of BX.

MCQ 6 (Conceptual)

What happens to register CX during each iteration of the loop?

- A. It is incremented
- B. It remains unchanged
- C. It is decremented by 1 ✓
- D. It is reset to 10

Explanation:

Each loop iteration subtracts 1 from CX to control the number of repetitions.

MCQ 7 (Conceptual)

Why is a memory variable named `total` declared?

- A. To store BX value
- B. To store CX value
- C. To store the final result of AX ✓
- D. To control the loop

Explanation:

After loop completion, the value in AX is stored in the memory variable **total**.

MCQ 8 (Conceptual)

Which instruction transfers the final result to memory?

- A. ADD
- B. SUB
- C. **MOV [total], AX** ✓
- D. JMP

Explanation:

The `MOV` instruction copies the value of AX into the memory variable `total`.

MCQ 9 (Conceptual)

What would happen if CX is initialized to 0?

- A. Loop executes infinitely
- B. Loop executes once
- C. **Loop does not execute at all** ✓
- D. Program crashes

Explanation:

Since `JNZ` checks for non-zero, a zero CX value causes the loop to be skipped.

MCQ 10 (Conceptual)

Why is `int 0x21` used at the end of the program?

- A. To start the loop
- B. To input data
- C. **To terminate the program properly** ✓
- D. To clear registers

Explanation:

`int 0x21` with `AX = 4C00h` is a DOS interrupt used to **terminate the program safely**.

🔍 LAB 3: Finding the Biggest Digit of VUID

Focus: Arrays, Memory, Comparison, Jumps, Subroutines

MCQ 1 (Single Correct)

What is the main objective of Lab 3?

- A. To sort digits of VUID
- B. To sum digits of VUID
- C. **To find the largest digit of VUID** ✓
- D. To reverse VUID

Explanation:

Lab 3 requires writing an assembly program that **finds the biggest digit** from the digits of a student's VUID.

MCQ 2 (Single Correct)

How are the digits of VUID stored in memory?

- A. As a string
- B. As a word array
- C. **As an array of digits (bytes)** ✓
- D. In registers only

Explanation:

The VUID digits are stored as a **byte array in memory** so that each digit can be accessed individually.

MCQ 3 (Single Correct)

Which variable stores the final biggest digit?

- A. array_nums
- B. total
- C. **max** ✓
- D. cx

Explanation:

A memory variable named **max** is declared to store the largest digit found.

MCQ 4 (Single Correct)

Which register is used as a pointer to traverse the array?

- A. AX
- B. CX
- C. **BX** ✓
- D. DX

Explanation:

The **BX register** is used to hold the address of the array and move through its elements.

MCQ 5 (Conceptual)

Which instruction is used to compare two digits?

- A. MOV
- B. ADD
- C. **CMP** ✓
- D. SUB

Explanation:

The **CMP** instruction compares two values and sets flags for conditional jumps.

MCQ 6 (Conceptual)

Which jump instruction is used when the current value is greater or equal?

- A. JL
- B. JNE
- C. **JGE** ✓
- D. JZ

Explanation:

JGE (Jump if Greater or Equal) is used to keep the current maximum value unchanged when it is already larger.

MCQ 7 (Conceptual)

Why are labels used in this program?

- A. To define variables
- B. To terminate program
- C. **To control program flow and looping ✓**
- D. To input data

Explanation:

Labels allow the program to **jump back and forth**, enabling looping and decision-making.

MCQ 8 (Conceptual)

What is the purpose of the `call` instruction in Lab 3?

- A. To end the program
- B. To jump without return
- C. **To execute a subroutine ✓**
- D. To load memory

Explanation:

`CALL` transfers control to a **subroutine** and returns back using `RET`.

MCQ 9 (Conceptual)

Why is `RET` used in the program?

- A. To compare values
- B. To repeat loop
- C. **To return control to the calling code ✓**
- D. To clear registers

Explanation:

`RET` returns execution from the subroutine to the main program.

MCQ 10 (Conceptual)

Why is `int 0x21` used at the end of the program?

- A. To print output
- B. To find maximum
- C. **To terminate the program properly** ✓
- D. To clear memory

Explanation:

`int 0x21` with termination service stops the program safely in DOS environment.

🔍 LAB 4: Summation of VUID Digits

Focus: Arrays, Looping, Indexing, Registers, Conditional Jumps

MCQ 1 (Single Correct)

What is the main objective of Lab 4?

- A. To find the largest digit of VUID
- B. To sort VUID digits
- C. **To calculate the sum of all digits of VUID** ✓
- D. To reverse VUID digits

Explanation:

Lab 4 focuses on **adding all digits** of the VUID and storing the result in a memory variable.

MCQ 2 (Single Correct)

In Lab 4, how are the digits of VUID stored?

- A. In registers only
- B. In stack
- C. **In a byte array in memory** ✓
- D. In a file

Explanation:

The VUID digits are stored as a **byte array** so that they can be accessed one by one using an index.

MCQ 3 (Single Correct)

Which register is used as the loop counter in Lab 4?

- A. AX
- B. BX
- C. **CX** ✓
- D. DX

Explanation:

CX holds the **number of digits** and is decremented in each loop iteration.

MCQ 4 (Single Correct)

Which register is used as an index to access array elements?

- A. BX
- B. SI
- C. **DI** ✓
- D. BP

Explanation:

DI (Destination Index) is used to point to successive memory locations of the VUID array.

MCQ 5 (Conceptual)

Which instruction adds each digit to the accumulator?

- A. MOV
- B. SUB
- C. **ADD AL, [vuid+DI]** ✓
- D. INC

Explanation:

This instruction adds the current digit from memory to register **AL**, which accumulates the sum.

MCQ 6 (Conceptual)

Why is **AL** used instead of **AX** to store the sum?

- A. AL is faster
- B. AX cannot store numbers
- C. **Digits are bytes, so AL is sufficient** ✓
- D. AX is reserved

Explanation:

Each digit is a **byte**, and their total sum fits within **8 bits**, so AL is enough.

MCQ 7 (Conceptual)

What is the purpose of the `DEC CX` instruction?

- A. To increase index
- B. To store sum
- C. **To control loop execution** ✓
- D. To reset registers

Explanation:

`DEC CX` decreases the loop counter and helps determine when the loop should stop.

MCQ 8 (Conceptual)

Which instruction causes the program to jump back to the loop label?

- A. `JMP`
- B. `JE`
- C. **`JNZ`** ✓
- D. `JZ`

Explanation:

`JNZ` jumps back to the loop label as long as CX is not zero.

MCQ 9 (Conceptual)

Where is the final sum stored?

- A. In AX
- B. In CX
- C. **In memory variable `sum`** ✓
- D. In DI

Explanation:

After completing the loop, the accumulated value in AL is stored in the memory variable **sum**.

MCQ 10 (Conceptual)

Why is `int 0x21` used at the end of the program?

- A. To display sum
- B. To repeat loop
- C. **To terminate the program safely** ✓
- D. To clear screen

Explanation:

`int 0x21` with termination service properly ends the program in DOS environment.

🔍 LAB 5: Bitwise Operations & Swapping Bit Pairs

Focus: Bit masking, AND, OR, ROL, ROR, Registers

MCQ 1 (Single Correct)

What is the main objective of Lab 5?

- A. To reverse bits of AX
- B. To shift bits left only
- C. **To swap every pair of bits in AX register** ✓
- D. To count number of bits

Explanation:

Lab 5 focuses on **swapping adjacent bit pairs** (e.g., bit 0 with 1, 2 with 3, etc.) in the AX register.

MCQ 2 (Single Correct)

Which register initially contains the binary form of the roll number?

- A. BX
- B. CX
- C. **AX** ✓
- D. DX

Explanation:

The binary value of the roll number is directly **loaded into AX**, which is the main working register.

MCQ 3 (Conceptual)

Why is the AND instruction used with masks like 0011001100110011b?

- A. To add bits
- B. To rotate bits
- C. **To isolate specific bit positions** ✓
- D. To clear AX completely

Explanation:

AND masking is used to **select specific bit pairs** while clearing the others.

MCQ 4 (Conceptual)

What is the purpose of `ROL BX, 2` in this program?

- A. To rotate bits right
- B. To clear BX
- C. **To move selected bit pairs to swapped positions** ✓
- D. To copy AX

Explanation:

After masking, `ROL BX, 2` shifts the selected bits left by two positions to help swap bit pairs.

MCQ 5 (Conceptual)

Why are the results in BX and DX combined using the OR instruction?

- A. To compare values
- B. To shift bits

- C. **To merge both swapped bit groups into one result** ✓
- D. To clear flags

Explanation:

Each register holds half of the swapped bits. **OR** merges them to form the final swapped-bit result.

🔍 LAB 6: Bubble Sort (Signed Numbers)

Focus: Arrays, Nested Loops, Comparisons, Conditional Jumps, Data Movement

MCQ 1 (Single Correct)

What is the main objective of Lab 6?

- A. To sort unsigned numbers
- B. To find the largest number
- C. **To sort an array of signed numbers using Bubble Sort** ✓
- D. To reverse an array

Explanation:

Lab 6 implements **Bubble Sort** to arrange **signed integers** (positive and negative) in order.

MCQ 2 (Conceptual)

Which instruction is used to correctly compare signed numbers?

- A. CMP with JA
- B. CMP with JB
- C. **CMP with JG / JL** ✓
- D. CMP with JZ

Explanation:

For **signed comparisons**, instructions like **JG (Jump if Greater)** and **JL (Jump if Less)** are used instead of unsigned jumps.

MCQ 3 (Single Correct)

Which register is commonly used as the array index in this lab?

- A. AX
- B. BX
- C. **SI** ✓
- D. DX

Explanation:

The **SI register** points to array elements and moves through memory during sorting.

MCQ 4 (Conceptual)

Why are two nested loops required in Bubble Sort?

- A. One loop for input, one for output
- B. One loop for comparison, one for printing
- C. **Outer loop for passes, inner loop for comparisons** ✓
- D. To reduce memory usage

Explanation:

Bubble Sort needs multiple **passes** over the array to ensure all elements are correctly ordered.

MCQ 5 (Conceptual)

What happens when two adjacent elements are out of order?

- A. Program terminates
- B. Values are ignored
- C. **They are swapped in memory** ✓
- D. Loop restarts

Explanation:

If the current element is greater than the next one, their **values are exchanged** to move larger values forward.

🔍 LAB 7: String Comparison / Manipulation

Focus: Strings, SI & DI registers, `CMPSB`, Flags, Conditional Jumps

MCQ 1 (Single Correct)

What is the main objective of Lab 7?

- A. To sort numbers
- B. To reverse an array
- C. **To compare two strings character by character** ✓
- D. To add string lengths

Explanation:

Lab 7 focuses on **string comparison**, where two strings are checked to see if they are equal or not.

MCQ 2 (Conceptual)

Which registers are used to point to the source and destination strings?

- A. AX and BX
- B. BX and CX
- C. **SI and DI** ✓
- D. DX and BP

Explanation:

SI (Source Index) points to the first string and **DI (Destination Index)** points to the second string.

MCQ 3 (Single Correct)

Which instruction compares bytes of two strings automatically?

- A. CMP
- B. MOVSB
- C. **CMPSB** ✓
- D. SCASB

Explanation:

CMPSB compares one byte from [SI] and [DI] and updates flags accordingly.

MCQ 4 (Conceptual)

What causes the comparison loop to terminate?

- A. When SI equals DI
- B. When CX becomes non-zero
- C. **When a mismatch is found or CX becomes zero** ✓
- D. When AX is cleared

Explanation:

The loop stops if **characters differ** or when all characters have been compared (CX = 0).

MCQ 5 (Conceptual)

Which flag is checked to determine if characters are equal?

- A. Carry Flag
- B. Sign Flag
- C. **Zero Flag** ✓
- D. Overflow Flag

Explanation:

If the **Zero Flag (ZF)** is set after `CMPSB`, the compared characters are equal.

🔗 LAB 8 – String Copy / Reverse

Focus: `MOVSB`, SI, DI, Direction Flag

MCQ 1

What is the purpose of Lab 8?

- A. Compare strings
- B. Count characters
- C. **Copy or reverse a string** ✓
- D. Sort strings

Explanation:

Lab 8 deals with **copying a string or reversing it** using string instructions.

MCQ 2

Which instruction copies one byte from source to destination?

- A. `CMPSB`
- B. `SCASB`
- C. **`MOVSB`** ✓
- D. `LODSB`

Explanation:

MOVSB copies a byte from [SI] to [DI].

MCQ 3

Which register points to source string?

- A. DI
- B. CX
- C. **SI** ✓
- D. BX

Explanation:

SI always points to the **source string**.

MCQ 4

Why is Direction Flag cleared (CLD)?

- A. To stop loop
- B. To reverse string
- C. **To move forward in memory** ✓
- D. To compare strings

Explanation:

CLD ensures SI and DI **increment forward**.

MCQ 5

Which register controls number of characters?

- A. AX
- B. BX
- C. **CX** ✓
- D. DX

Explanation:

CX holds the **string length counter**.

🔗 LAB 9 – Searching a Character in a String

Focus: SCASB, ZF, Looping

MCQ 1

Main objective of Lab 9 is to:

- A. Copy string

- B. Reverse string
- C. **Search a character in string** ✓
- D. Compare strings

Explanation:

Lab 9 searches for a **specific character** in a string.

MCQ 2

Which instruction scans a string byte?

- A. MOVSB
- B. CMPSB
- C. **SCASB** ✓
- D. LODSB

Explanation:

SCASB compares AL with memory byte.

MCQ 3

Which flag indicates character found?

- A. CF
- B. SF
- C. **ZF** ✓
- D. OF

Explanation:

ZF = 1 means **match found**.

MCQ 4

Which register holds search character?

- A. BX
- B. CX
- C. **AL** ✓
- D. DX

Explanation:

Character to search is stored in **AL**.

MCQ 5

Why CX is used?

- A. Store address
- B. Store character

C. **Control loop iterations** ✓

D. Store result

Explanation:

CX counts remaining characters.

🔗 LAB 10 – Count Number of Vowels

Focus: Comparison, Loop, Conditional Jumps

MCQ 1

Objective of Lab 10 is to:

A. Sort string

B. Reverse string

C. **Count vowels in string** ✓

D. Copy string

Explanation:

Program counts vowels like A, E, I, O, U.

MCQ 2

Which instruction compares characters?

A. MOV

B. ADD

C. **CMP** ✓

D. INC

Explanation:

CMP compares characters with vowels.

MCQ 3

Which register points to string?

A. BX

B. CX

C. **SI** ✓

D. DX

Explanation:

SI traverses string characters.

MCQ 4

Where is vowel count stored?

- A. AX
- B. BX
- C. **Memory variable** ✓
- D. DX

Explanation:

Final count is saved in memory.

MCQ 5

Why conditional jumps are used?

- A. End program
- B. Copy data
- C. **Check vowel match** ✓
- D. Clear registers

Explanation:

They decide whether to increment count.

? LAB 11 – Stack Operations

Focus: PUSH, POP, LIFO

MCQ 1

Main objective of Lab 11 is to:

- A. Sort data
- B. **Understand stack operations** ✓
- C. Copy string
- D. Compare data

Explanation:

Lab 11 demonstrates **stack usage**.

MCQ 2

Stack follows which principle?

- A. FIFO
- B. LILO
- C. **LIFO** ✓
- D. FILO

Explanation:

Last In, First Out.

MCQ 3

Which instruction pushes data onto stack?

- A. POP
- B. MOV
- C. **PUSH** ✓
- D. ADD

Explanation:

PUSH stores data on stack.

MCQ 4

Which instruction removes data from stack?

- A. PUSH
- B. ADD
- C. MOV
- D. **POP** ✓

Explanation:

POP retrieves data.

MCQ 5

Which register controls stack top?

- A. BP
- B. BX
- C. **SP** ✓
- D. SI

Explanation:

SP points to top of stack.

[🔗 LAB 12 - Procedure / Subroutine](#)

Focus: CALL, RET, Stack

MCQ 1

Purpose of Lab 12 is to:

- A. Loop creation

- B. Sorting
- C. **Use procedures (subroutines)** ✓
- D. String copy

Explanation:

Lab 12 explains modular programming.

MCQ 2

Which instruction calls a procedure?

- A. JMP
- B. **CALL** ✓
- C. LOOP
- D. RET

Explanation:

CALL jumps to subroutine.

MCQ 3

Which instruction returns from procedure?

- A. CALL
- B. JMP
- C. **RET** ✓
- D. POP

Explanation:

RET returns execution.

MCQ 4

Where is return address stored?

- A. Register
- B. Memory variable
- C. **Stack** ✓
- D. Flag

Explanation:

CALL stores return address on stack.

MCQ 5

Main advantage of procedures is:

- A. Faster execution
- B. Less memory

C. **Code reusability** ✓

D. No loops

Explanation:

Procedures avoid code repetition.

🔖 LAB 13 – File Handling (Read / Write Operations)

Focus: Files, DOS interrupts, File modes

MCQ 1

What is the main purpose of Lab 13?

A. Stack operations

B. String comparison

C. **File handling (read/write)** ✓

D. Sorting data

Explanation:

Lab 13 introduces **file creation, reading, and writing** using DOS services.

MCQ 2

Which interrupt is used for file handling in assembly?

A. int 10h

B. int 13h

C. **int 21h** ✓

D. int 33h

Explanation:

int 21h provides DOS services including **file operations**.

MCQ 3

Which register stores the file handle?

A. BX

B. CX

C. **AX** ✓

D. DX

Explanation:

DOS returns the **file handle in AX** after opening/creating a file.

MCQ 4

Which mode opens a file for reading only?

- A. 01h
- B. **00h** ✓
- C. 02h
- D. 03h

Explanation:

Mode 00h is used for **read-only access**.

MCQ 5

Why must a file be closed after use?

- A. To clear registers
- B. To restart program
- C. **To free system resources** ✓
- D. To display output

Explanation:

Closing files prevents **memory/resource leakage**.

🔗 LAB 14 - Interrupts and Keyboard Input

Focus: Keyboard input, interrupts, ASCII

MCQ 1

Main objective of Lab 14 is to:

- A. Sort data
- B. File reading
- C. **Handle keyboard input using interrupts** ✓
- D. Stack usage

Explanation:

Lab 14 demonstrates **keyboard input handling**.

MCQ 2

Which interrupt reads a key from keyboard?

- A. int 10h
- B. **int 21h** ✓
- C. int 13h
- D. int 80h

Explanation:

Keyboard input is taken via **int 21h services**.

MCQ 3

Which register holds the ASCII value of key pressed?

- A. BX
- B. CX
- C. **AL** ✓
- D. DX

Explanation:

The input character is returned in **AL**.

MCQ 4

Why ASCII conversion is required?

- A. For memory access
- B. For loops
- C. **To convert characters to numeric values** ✓
- D. For stack operations

Explanation:

Keyboard input is ASCII; conversion is needed for calculations.

MCQ 5

Which instruction compares user input?

- A. MOV

- B. ADD
- C. **CMP** ✓
- D. INC

Explanation:

CMP is used to compare input with expected value.

🔗 LAB 15 - Arithmetic Operations (MUL / DIV)

Focus: Multiplication, Division, Registers

MCQ 1

Main objective of Lab 15 is to:

- A. Copy strings
- B. File handling
- C. **Perform multiplication and division** ✓
- D. Stack handling

Explanation:

Lab 15 covers **arithmetic instructions**.

MCQ 2

Which instruction performs unsigned multiplication?

- A. IMUL
- B. ADD
- C. **MUL** ✓
- D. DIV

Explanation:

MUL performs **unsigned multiplication**.

MCQ 3

Which instruction is used for signed multiplication?

- A. MUL
- B. **IMUL** ✓

- C. DIV
- D. ADD

Explanation:

IMUL handles **signed numbers**.

MCQ 4

In division, where is the quotient stored?

- A. BX
- B. CX
- C. **AX** ✓
- D. DX

Explanation:

After DIV, **quotient is stored in AX**.

MCQ 5

Where is remainder stored after division?

- A. AX
- B. CX
- C. **DX** ✓
- D. BX

Explanation:

Remainder is stored in **DX**.

🔗 LAB 16 – Complete Program Integration

Focus: Loops, Procedures, Strings, Arithmetic

MCQ 1

Main purpose of Lab 16 is to:

- A. Learn one instruction
- B. Use stack only
- C. **Integrate multiple assembly concepts** ✓
- D. File deletion

Explanation:

Lab 16 combines **loops, procedures, strings, arithmetic.**

MCQ 2

Which concept improves program modularity?

- A. Loops
- B. Interrupts
- C. **Procedures** ✓
- D. Flags

Explanation:

Procedures make code **modular and reusable.**

MCQ 3

Which instruction returns control to OS?

- A. RET
- B. JMP
- C. **int 21h** ✓
- D. CALL

Explanation:

Program terminates using **int 21h service.**

MCQ 4

Why registers must be managed carefully?

- A. To reduce loops
- B. To increase speed
- C. **To avoid data corruption** ✓
- D. To save memory

Explanation:

Incorrect register handling causes **wrong results.**

MCQ 5

Which skill is most important in Lab 16?

- A. Typing speed
- B. Syntax memorization
- C. **Logical program flow understanding** ✓
- D. Printing output

Explanation:

Integration requires **strong logical understanding**.

THE END

VU

Updates Tech