MASM 8086 Tutorial: INT 21h, String Handling, Number Parsing, and Output

1. Different INT 21h Functions

INT 21h is the DOS interrupt used for input, output, and system services in 8086 assembly. The function number is placed in AH before executing INT 21h.

- 01h Input single character with echo. Character returned in AL.
- 02h Output single character. Character must be in DL.
- 07h Input single character without echo.
- 08h Input single character without echo with Ctrl C check.
- 09h Display string. String must end with $.
- 0Ah Buffered string input using defined buffer structure.
- 4Ch Terminate program and return control to DOS.
- 3Ch Create file.
- 3Dh Open file.
- 3Eh Close file.
- 3Fh Read file.
- 40h Write file.

2. Conventional Ways of Fetching Input

There are two conventional approaches for fetching input in 8086 using DOS services.

Buffered Input Using Function 0Ah

```
Buffer Definition:

buffer  DB 20
        DB ?
        DB 20 DUP(?)

Reading Input:

MOV AH, 0Ah
LEA DX, buffer
INT 21h
```

Byte by Byte Input Using Function 01h

```
MOV AH, 01h
INT 21h
MOV BL, AL
```

3. ASCII Input and Case Conversion

```
Concept:

Keyboard input is stored as ASCII values.
'A' to 'Z' range from 41h to 5Ah.
'a' to 'z' range from 61h to 7Ah.
Lowercase letters are 20h greater than uppercase letters.
To convert uppercase to lowercase add 20h.
To convert lowercase to uppercase subtract 20h.


Example Code:

CMP AL, 'a'
JB skip
CMP AL, 'z'
JA skip
SUB AL, 20h
skip:


Practice Questions:

1. Write a program that converts a full string to uppercase.
2. Write a program that toggles case of each alphabet in input.
3. Count number of lowercase characters in a string.
```

4. ASCII to Numbers and Extracting Numbers from Input

```
Concept:

Digits typed from keyboard are ASCII characters.
'0' equals 30h.
'9' equals 39h.

To convert ASCII digit to numeric value:
SUB AL, '0'

To build multi digit number:
number = number * 10 + digit


Example Code:

SUB AL, '0'
MOV AH, 0
MOV BX, 10
MUL BX
ADD AX, digit
```

5. Output Techniques and Converting Numbers to ASCII

```
Concept:
To display characters or strings, ASCII values must be sent to DOS services.
```

To display a string:
Use function 09h and terminate string with $.

To display single character:
Use function 02h with character in DL.

To display a number:
Convert numeric value to ASCII before displaying.
This is done by repeatedly dividing by 10, pushing remainders, then adding '0' to each digit before p

Display String:

```
MOV AH, 09h
LEA DX, message
INT 21h
```

Display Character:

```
MOV DL, 'A'
MOV AH, 02h
INT 21h
```

Number to ASCII Conversion Pattern:

```
MOV CX, 0
MOV BX, 10

convert_loop:
XOR DX, DX
DIV BX
PUSH DX
INC CX
CMP AX, 0
JNE convert_loop

print_loop:
POP DX
ADD DL, '0'
MOV AH, 02h
INT 21h
LOOP print_loop
```