



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2024-25

Class:	SE	Semester:	IV
Course Code:	CSL404	Course Name:	Microprocessor Lab

Name of Student:	Bhagyashri Kaleni Sutar
Roll No. :	75
Experiment No.:	6
Title of the Experiment:	Program to reverse word in a string
Date of Performance:	17/02/2025
Date of Submission:	24/02/2025

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Performance	5	
Understanding	5	
Journal work and timely submission	10	
Total	20	

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Performance	4-5	2-3	1
Understanding	4-5	2-3	1
Journal work and timely submission	8-10	5-8	1-4

Checked by

Name of Faculty : Ms. Sweety Patil

Signature :

Date:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Aim: Assembly Language Program to reverse the word in string.

Theory:

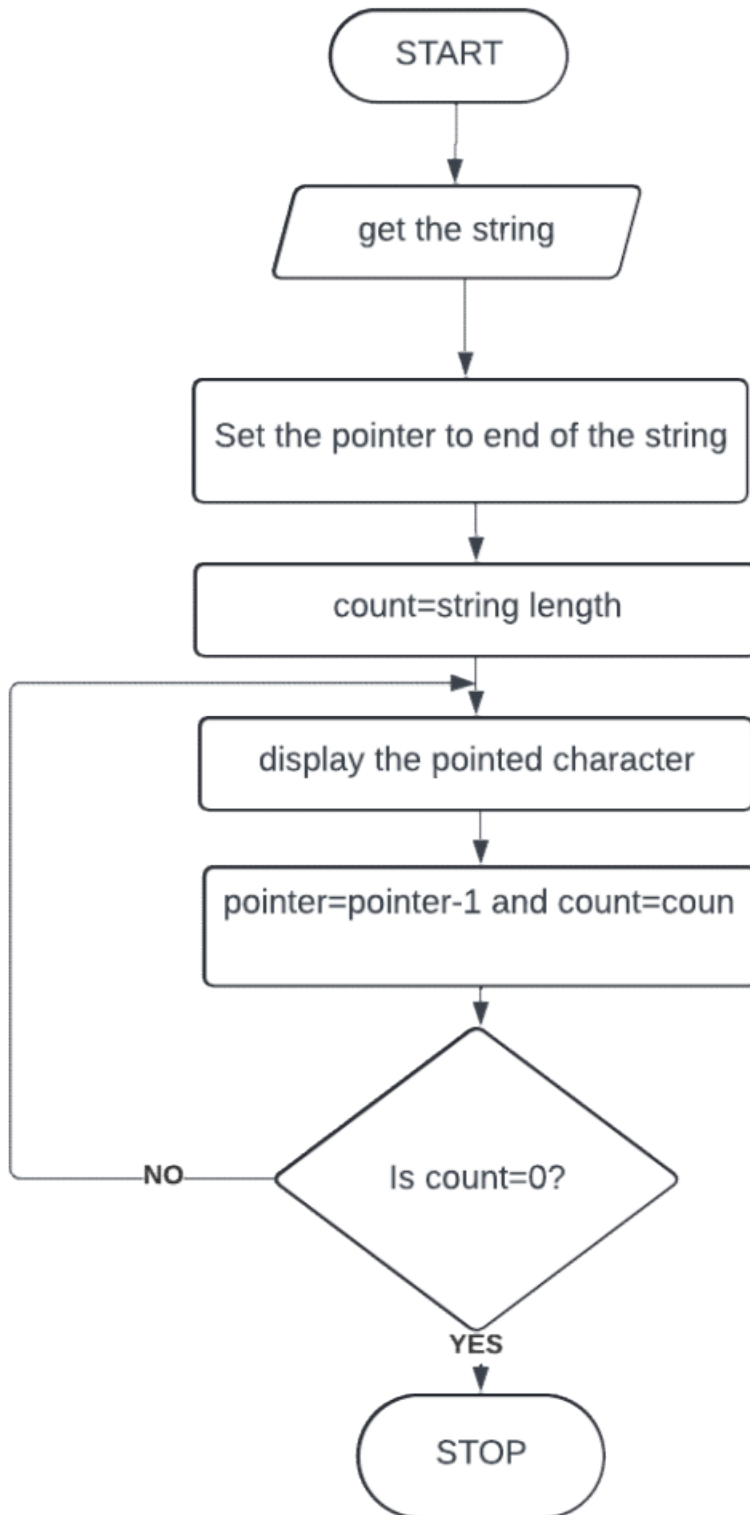
This program will read the string entered by the user and then reverse it. Reverse a string is the technique that reverses or changes the order of a given string so that the last character of the string becomes the first character of the string and so on.

Algorithm:

1. Start
2. Initialize the data segment
3. Display the message -1
4. Input the string
5. Display the message 2
6. Take characters count in DI
7. Point to the end character and read it
8. Display the character
9. Decrement the count
10. Repeat until the count is zero
11. To terminate the program using DOS interrupt
 - a. Initialize AH with 4ch
 - b. Call interrupt INT 21h
12. Stop



Flowchart:





Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Code:

```
.org data
.data
m1 db 10,13,'Enter Sting in UPPERCASE:$'
m2 db 10,13,'REVERSE String is:$'
buff db 88

.code
lea dx, m1
mov ah, 09h
int 21h

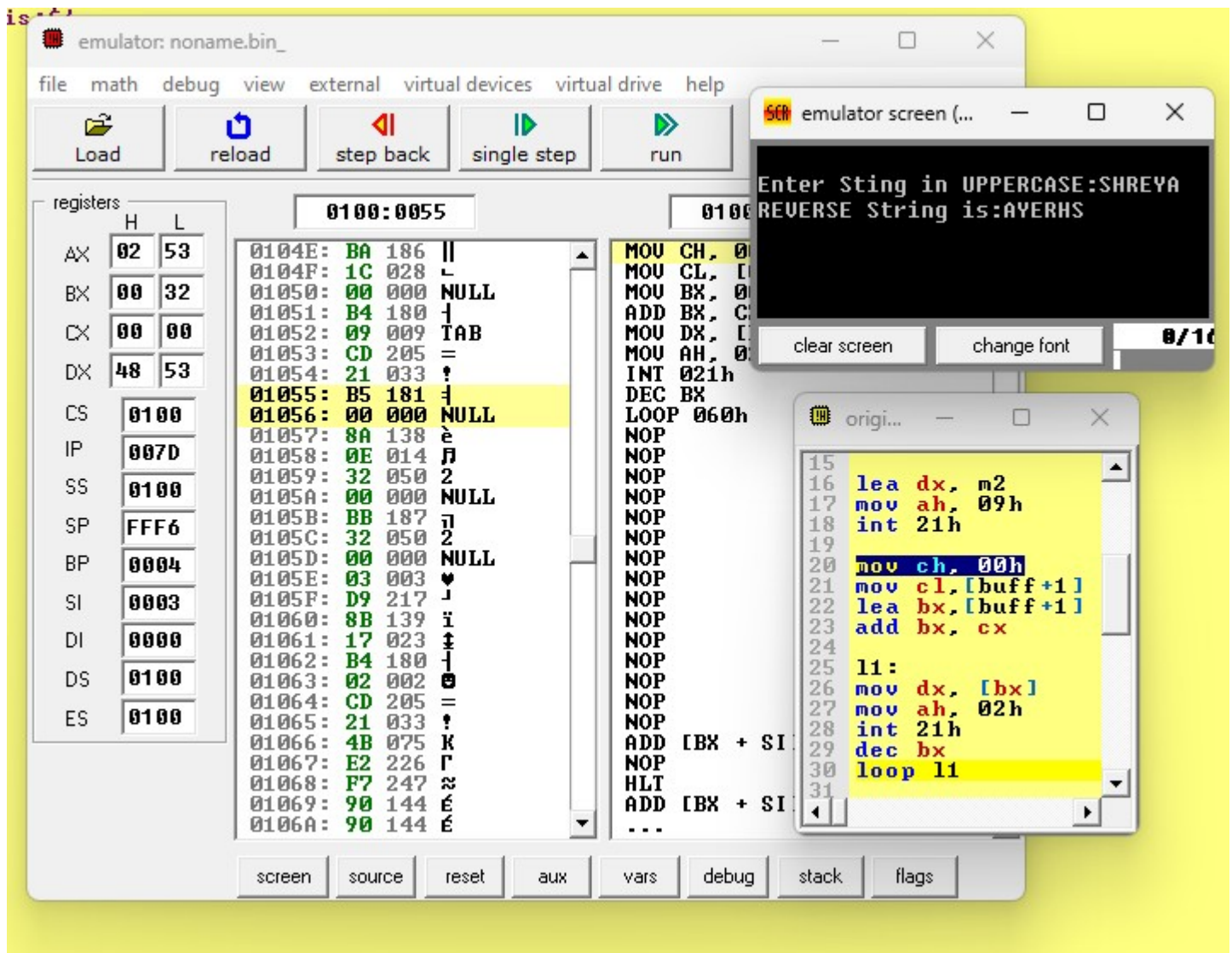
lea dx, buff
mov ah, 0ah
int 21h

lea dx, m2
mov ah, 09h
int 21h

mov ch, 00h
mov cl, [buff+1]
lea bx, [buff+1]
add bx, cx

l1:
mov dx, [bx]
mov ah, 02h
int 21h
dec bx
loop l1
```

Output:



Conclusion:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

1. Explain the difference between XLAT and XLATB

Ans:-

Feature	XLAT	XLATB
Opcode	XLAT	XLATB
Base Segment Register	DS (Data Segment)	ES (Extra Segment)
Table Location	The lookup table starts at the address pointed to by DS:BX.	The lookup table starts at the address pointed to by ES:BX.
Index Register	Uses the value in AL as the index into the table.	Uses the value in AL as the index into the table.
Result	The byte at the index in the table is copied to AL.	The byte at the index in the table is copied to AL.
Common Use Case	Commonly used in data segments (DS).	Less common, used for extra data or special segments (ES).
Effect on AL	Replaces the value in AL with the byte found at DS:BX + AL.	Replaces the value in AL with the byte found at ES:BX + AL.
General Purpose	More commonly used in x86 assembly.	Less frequently used, more specific use cases.

2. Explain the instruction LAHF

Ans:-1. LAHF (Load AH from Flags):

- **Opcode:** LAHF
- **Function:** This instruction loads the lower byte of the Flags register (the status flags) into the AH register.
- **Description:**
 - It stores the lower 8 bits of the flag register (Flags register) into the AH register.
 - These 8 bits include the **sign flag**, **zero flag**, **auxiliary carry flag**, **parity flag**, and **carry flag**.
 - This is typically used to preserve or examine the status flags at a later point in the program

Example.

LAHF ; Loads the lower 8 bits of the flag register into AH



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

2. LAX (Load and Exchange):

- If you meant **LAX** or any other instruction, let me know and I can explain that. However, **LAH** as an instruction is not recognized in typical assembly or the x86 instruction set.

