AIDS MICROPROCESSOR LAB S21 BATCH (2023-24)

Experiment 7(a) Title: Assembly language programming based on String operation.

Name of student: Meet Raut Class Roll Number: 2201084

Date of Performance: 01/04/2024

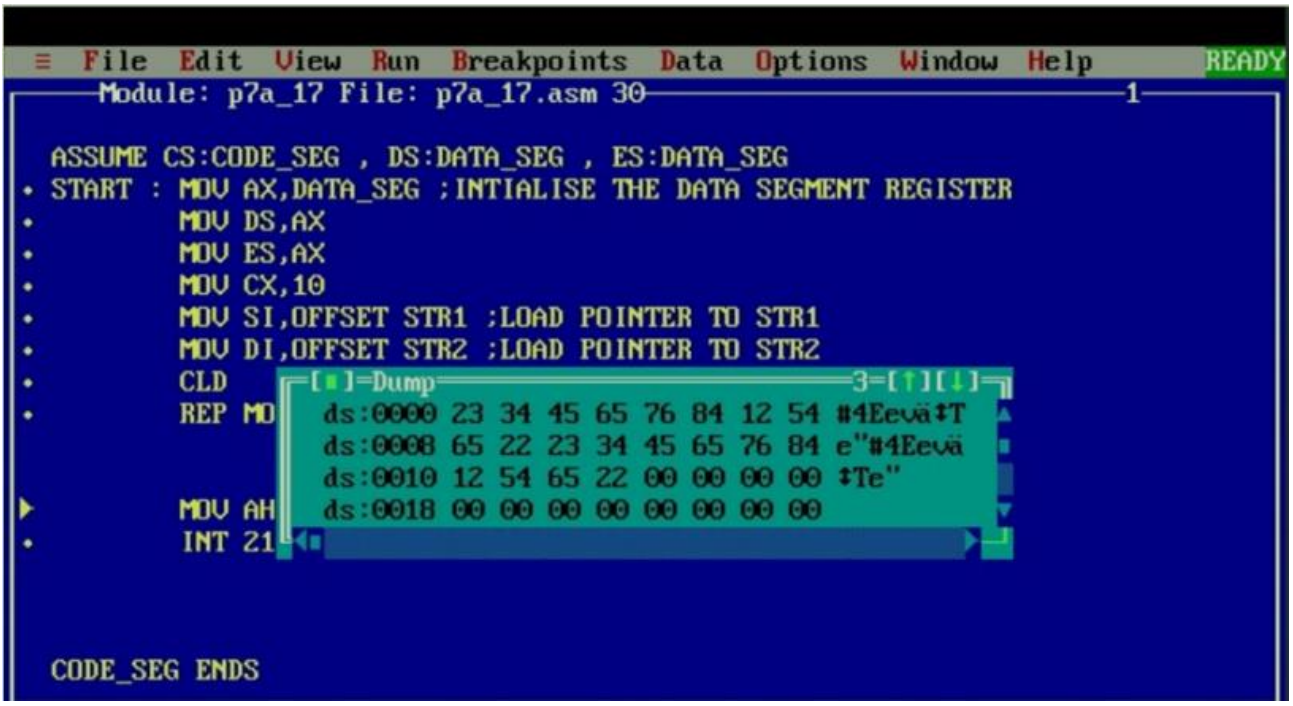
Batch: S2-1 Timing: 3:00-5:00 Date of Submission: 01/04/2024

Assembly language code

```
DATA_SEG SEGMENT
    STR1 DB 23H,34H,45H,65H,76H,84H,12H,54H,65H,22H
    STR2 DB 10 DUP(0)
DATA_SEG ENDS

CODE_SEG SEGMENT
ASSUME CS:CODE_SEG , DS:DATA_SEG , ES:DATA_SEG
START :
MOV AX,DATA_SEG              #INTIALISE THE DATA SEGMENT REGISTER
MOV DS,AX
MOV ES,AX
MOV CX,10
MOV SI,OFFSET STR1                  #LOAD POINTER TO STR1
MOV DI,OFFSET STR2                  #LOAD POINTER TO STR2
CLD                              #CLEAR DIRECTION FLAG
REP MOVSB                         #MOVE ONE BYTE
MOV AH,4CH                     #REQUEST TO TERMINATE
INT 21H                          #EXIT TO DOS
CODE_SEG ENDS
END START
```

Result:


File Edit View Run Breakpoints Data Options Window Help READY
Module: p7a_17 File: p7a_17.asm 30
ASSUME CS:CODE_SEG , DS:DATA_SEG , ES:DATA_SEG
START : MOV AX,DATA_SEG ;INTIALISE THE DATA SEGMENT REGISTER
MOV DS,AX
MOV ES,AX
MOV CX,10
MOV SI,OFFSET STR1 ;LOAD POINTER TO STR1
MOV DI,OFFSET STR2 ;LOAD POINTER TO STR2
CLD
REP MO
MOV AH
INT 21
Dump
ds:0000 23 34 45 65 76 84 12 54 #4Eevä↕T
ds:0008 65 22 23 34 45 65 76 84 e"#4Eevä
ds:0010 12 54 65 22 00 00 00 00 ↕Te"
ds:0018 00 00 00 00 00 00 00 00
CODE_SEG ENDS

Experiment 7(b) Title: Assembly language programming based on String operation.

Name of student: Meet Raut Class Roll Number: 2201084

Date of Performance: 01/04/2024

Batch: S2-1 Timing: 3:00-5:00 Date of Submission: 01/04/2024

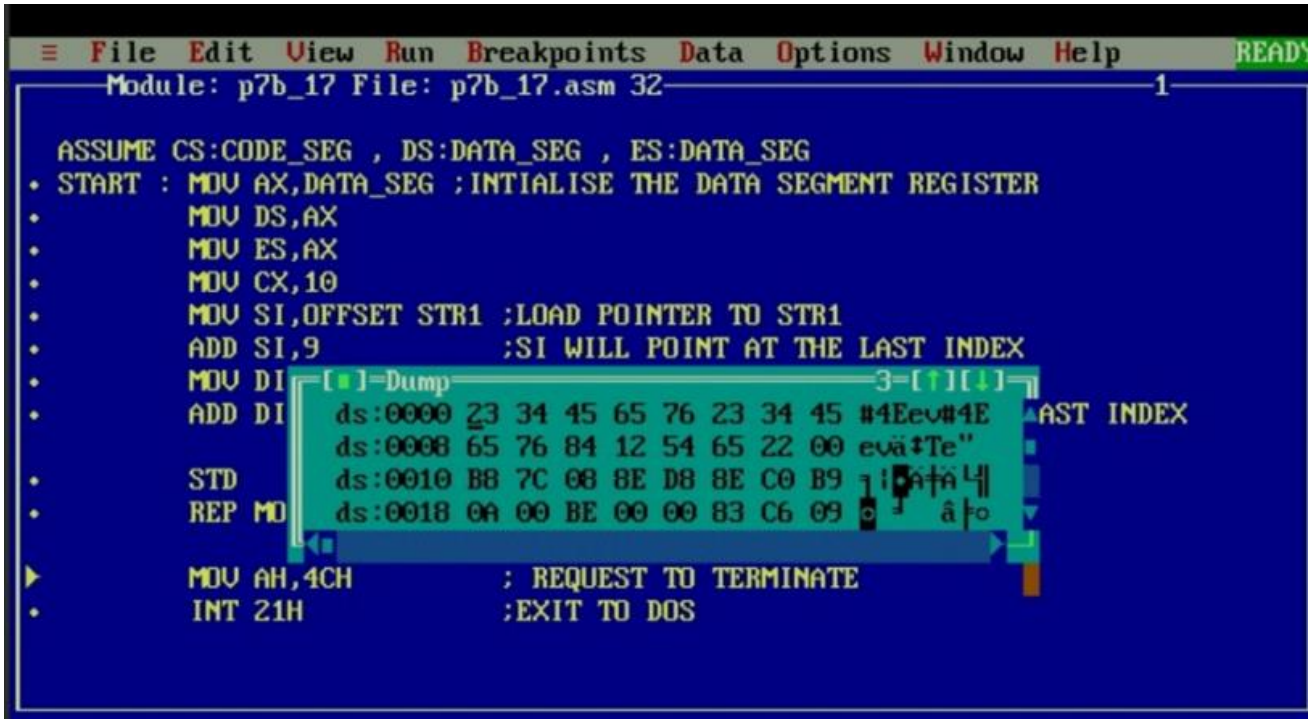
Assembly language code

```
DATA_SEG SEGMENT
    STR1 DB 23H,34H,45H,65H,76H,84H,12H,54H,65H,22H
DATA_SEG ENDS
CODE_SEG SEGMENT

ASSUME CS:CODE_SEG , DS:DATA_SEG , ES:DATA_SEG
START :
MOV AX,DATA_SEG                 #INTIALISE THE DATA SEGMENT REGISTER
MOV DS,AX
MOV ES,AX
MOV CX,10
MOV SI,OFFSET STR1              #LOAD POINTER TO STR1
ADD SI,9                        #SI WILL POINT AT THE LAST INDEX
MOV DI,SI                       #DI WILL POINT AT THE LAST INDEX
ADD DI,5                        #DI WILL POINT TO 5 INDEX AFTER LAST INDEX
STD
REP MOVSB
MOV AH,4CH                      #REQUEST TO TERMINATE
INT 21H                         #EXIT TO DOS

CODE_SEG ENDS
END START
```

Result:



CONCLUSION: *LO 2, LO 3 mapped.*
