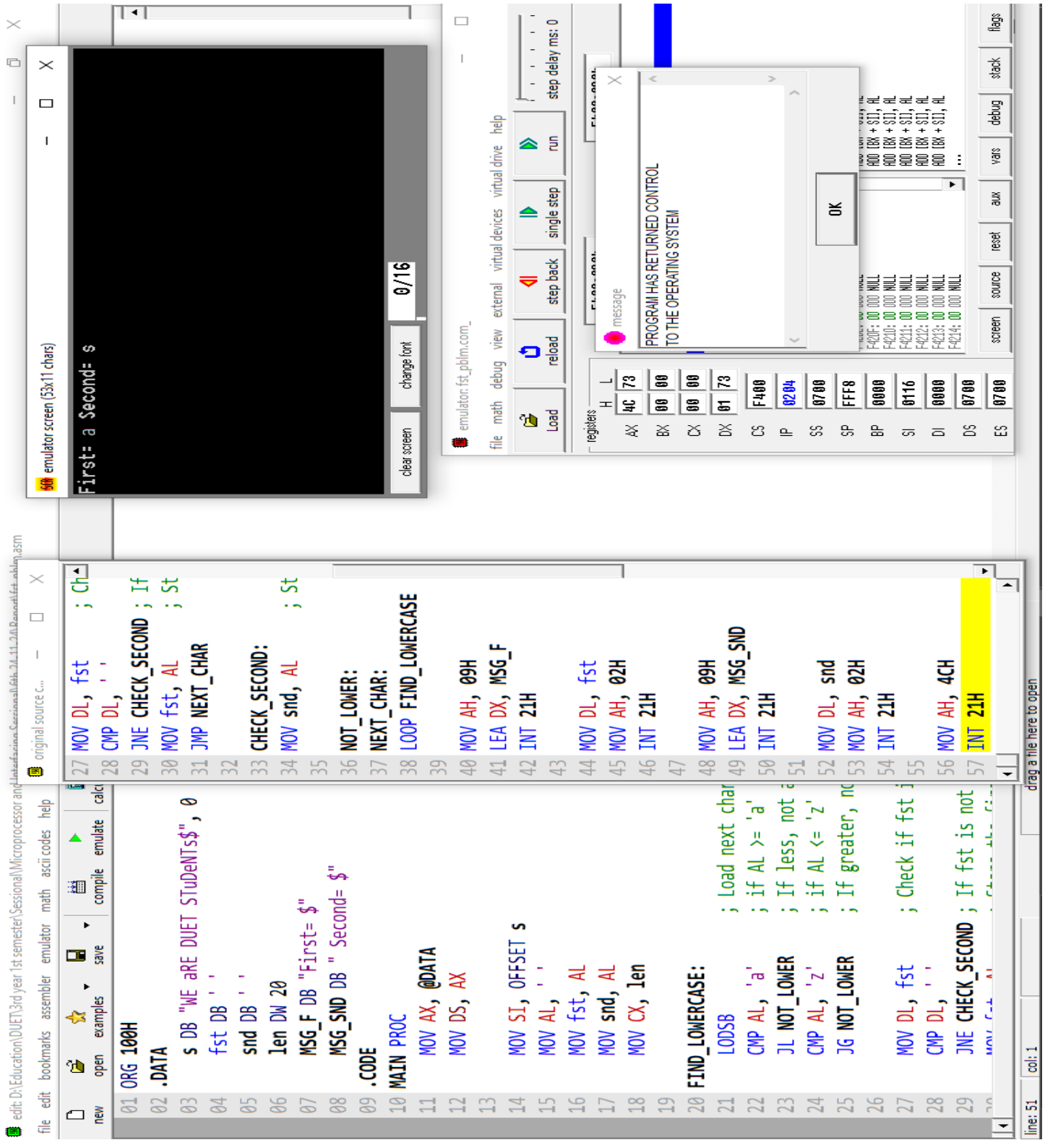
1) Write an assembly language program that stores a string in a variable. Now, first display the whole string and then display the first small letter and last small letter in the string. If no small letters are entered, then display “No small letters”.

Sample Input / Output: Input in a String: input_string DB ‘WE aRE DUET STuDeNTs’, 0Dh, 0Ah, ‘$’

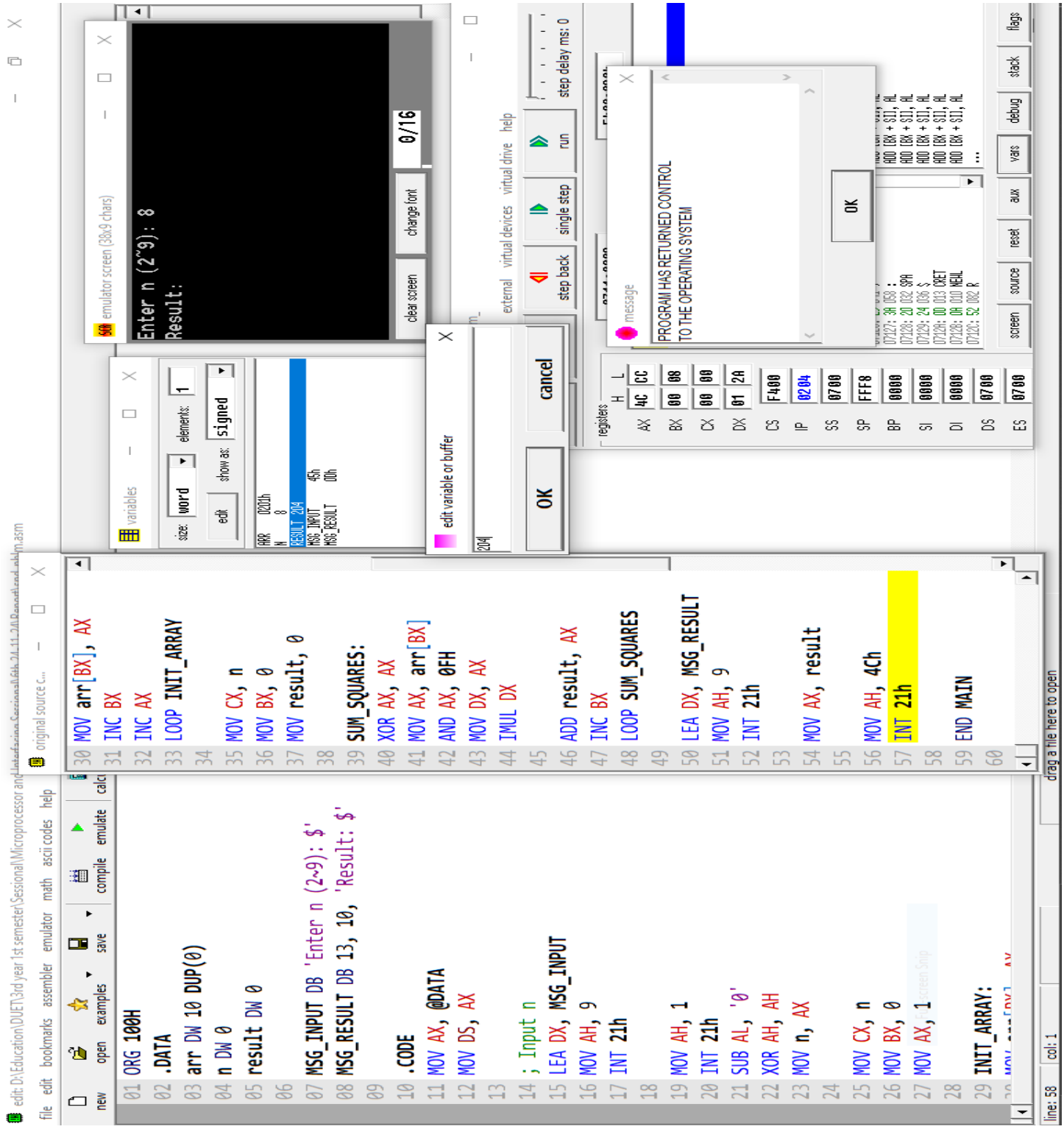
Output: a

S

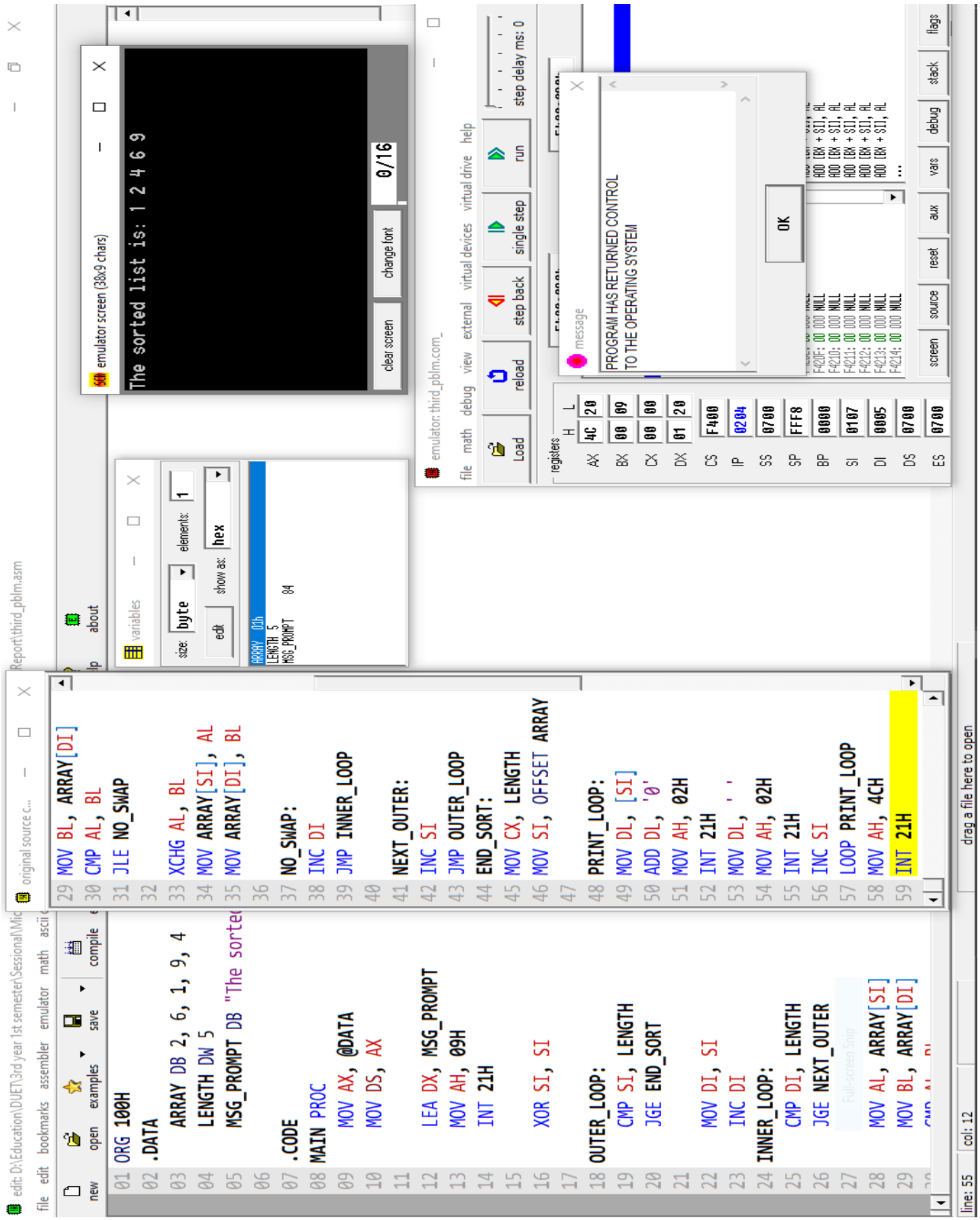


2) Write an assembly language code to derive the final value of the number sequence 1 2+22+32+42+…..+N2 . (use ARRAY and Loop). Take the input value of N (in between 2 to 9) as a single ASCII character and then adjust it to actual decimal value in your program. Finally, store and show the output in a variable named RESULT.

Sample Input / Output:
Input: The value of N in between 2 ~9 : 9
The result is: 285



3) Write an assembly code to sort the following data in ascending order using any sorting algorithm.
Sample Input: Sample 2 6 1 9 4 Output: The sorted list is: 1 2 4 6 9



Q1: The user will input 3 numbers (single digit) and the program will output the second highest number. The sample input and output is given below:

Sample Input	Sample Output
9, 5, 8	8
9, 9, 6	6
9, 9, 9	All the numbers are equal
9, 6, 6	6

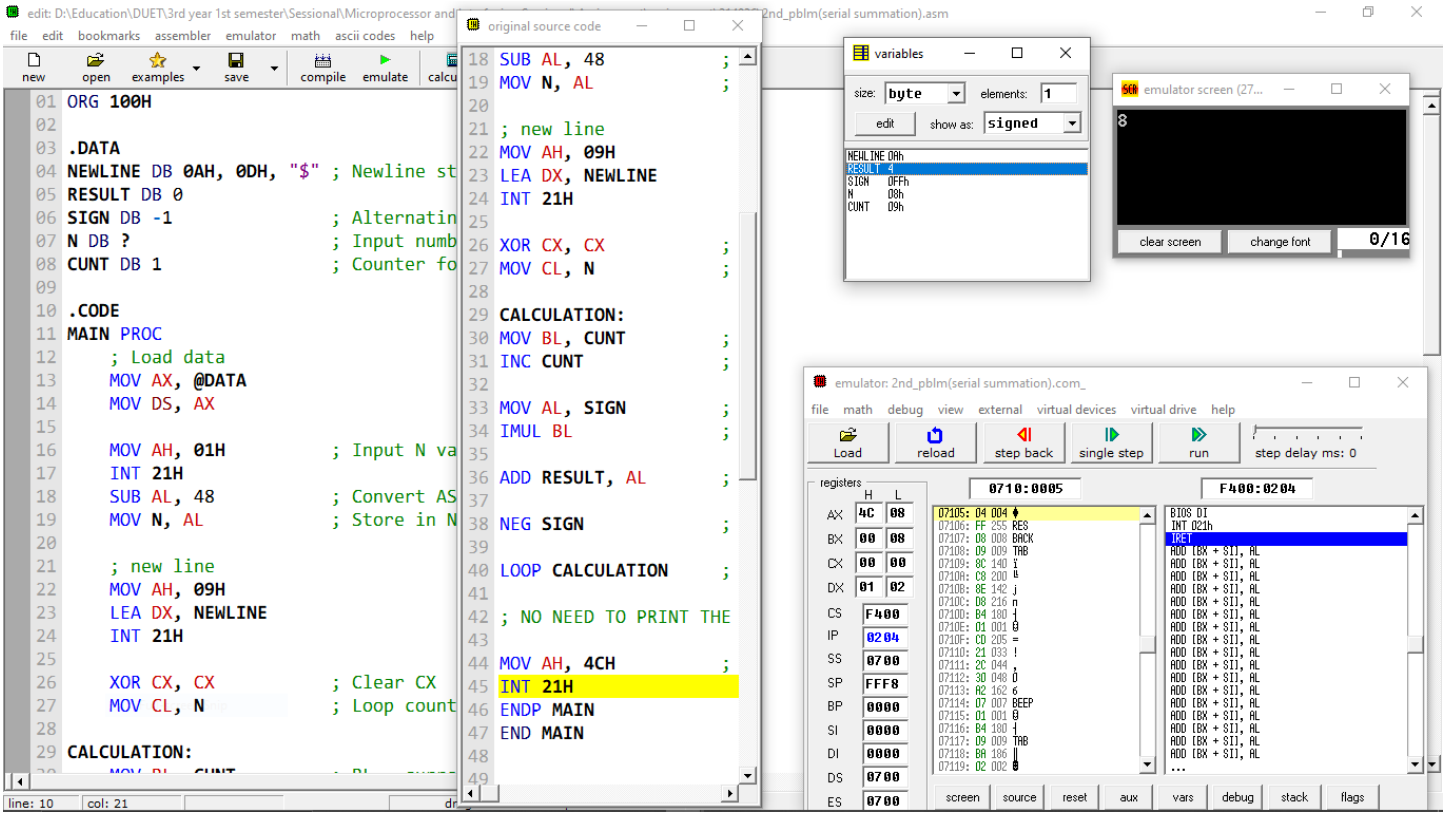
```
ORG 100H

.DATA
EQUAL DB "All the numbers are equal$
NEWLINE DB 0AH, 0DH, "$"
A DB ?
B DB ?
C DB ?
FST DB 0
SND DB 0

.CODE

MAIN PROC
    ; DATA LOAD
    MOV AX, @DATA
    MOV DS, AX

    ; THREE INPUT A, B, C VARIABLE
    MOV AH, 01H
    INT 21H
    MOV A, AL

    INT 21H
    MOV B, AL

    INT 21H
    MOV C, AL
```

```
; NEW LINE
MOV AH, 09H
LEA DX, NEWLINE
INT 21H

MOV BL, A
MOV FST, BL       ; FST

; CHECK EQUAL OR NOT

MOV BL, B
CMP BL, A         ; A
JE A_B_EQUAL      ; YE
JMP FIND_SND      ; 3

A_B_EQUAL:
MOV BL, B
CMP BL, C
JE ABC_EQUAL  ; A == B
JMP FIND_SND  ;


ABC_EQUAL:
MOV AH, 09H
LEA DX, EQUAL  ; PRINT
INT 21H
JMP EXIT

FIND_SND:
MOV BL, B
CMP BL, FST
JLE CHK_FSTE_C
```

```
CMP BL, FST
JLE CHK_FSTE_C
MOV BL, FST          ;
MOV SND, BL          ;
MOV BL, B
MOV FST, BL


CHK_FSTE_C:
MOV BL, C
CMP BL, FST
JLE DOUBLE_CHK
MOV BL, FST          ; S
MOV SND, BL          ; F
MOV BL, B
MOV FST, BL

DOUBLE_CHK:
MOV BL, A
CMP BL, FST
JGE CHK_SND_B
MOV BL, A
CMP BL, SND
JLE CHK_SND_B
MOV BL, A
MOV SND, BL

CHK_SND_B:
MOV BL, B
CMP BL, FST
JGE CHK_SND_C
MOV BL, B
```

```
JGE CHK_SND_C
MOV BL, B
CMP BL, SND
JLE CHK_SND_C
MOV BL, B
MOV SND, BL

CHK_SND_C:
MOV BL, C
CMP BL, FST
JGE OUTPUT
MOV BL, C
CMP BL, SND
JLE OUTPUT
MOV BL, C
MOV SND, BL

; OUTPUT SHOW
OUTPUT:
MOV AH, 02H
MOV DL, SND
INT 21H
JMP EXIT

EXIT:
MOV AH, 4CH
INT 21H

ENDP MAIN
END MAIN
RET
```

Q2. Write an assembly language code to derive the final value of the number sequence: -1+2-3+4-5+.....N. (Using Loop). Take the input value of N (in between 2 to 9) as a single ASCII character and then adjust it to the actual decimal value in your program. Finally, store the output in a variable named RESULT. You do not need to display the output in the console



Q3: First of all we will input a mark of the subject from the candidate and according to the following condition we will calculate the grade.

Sample Input: 80 Ouput: Grade: A