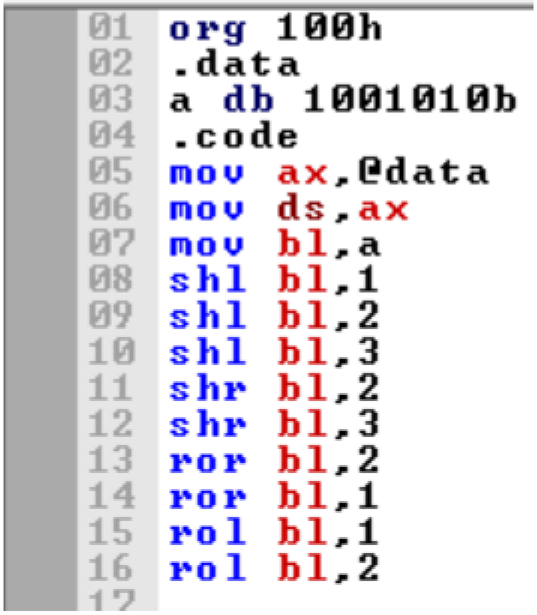
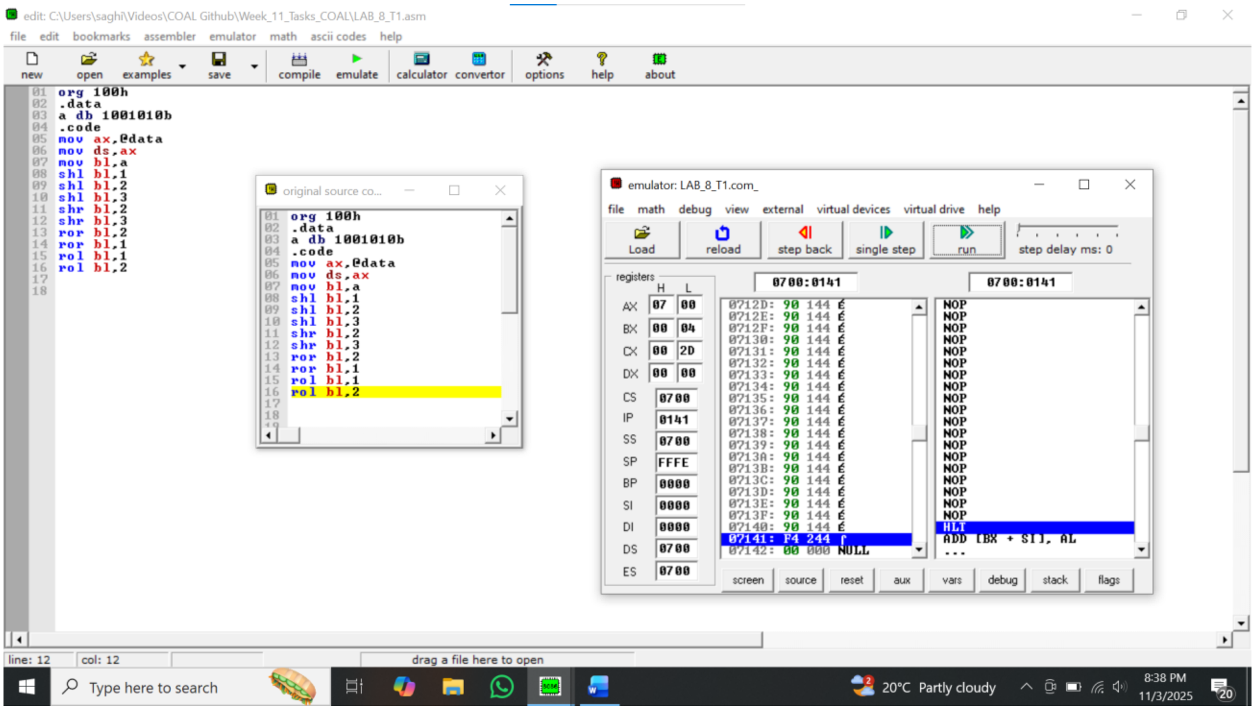
LAB – 08 – Week – 11 – 67005 – Saghir Ali – COAL

Task 01 :

It performs a workout of bit shifts and rotations on a binary number. The value in the register is moved left, right, and rotated in circles, completely transforming the original pattern by the end.



```
org 100h
.data
a db 10010010b
.code
mov ax,@data
mov ds,ax
mov bl,a
shl bl,1
shl bl,2
shl bl,3
shr bl,2
shr bl,3
ror bl,2
ror bl,1
rol bl,1
rol bl,2
```



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T1.asm

```
org 100h
.data
a db 1001010b
.code
mov ax,@data
mov ds,ax
mov bl,a
shl bl,1
shl bl,2
shl bl,3
shr bl,2
shr bl,3
ror bl,2
ror bl,1
rol bl,1
rol bl,2
```

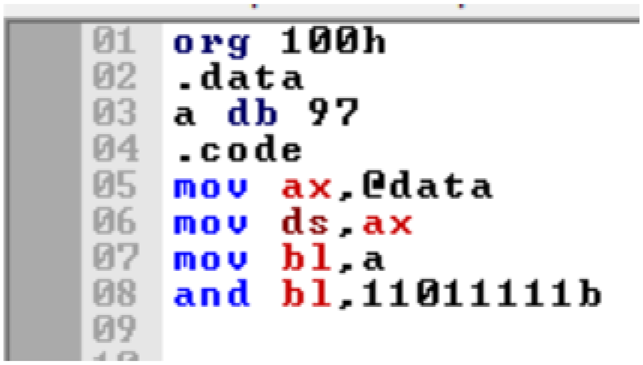
emulator: LAB_8_T1.com_

```
registers
     H   L
AX   07  00
BX   00  04
CX   00  2D
DX   00  00
CS   0700
IP   0141
SS   0700
SP   FFFE
BP   0000
SI   0000
DI   0000
DS   0700
ES   0700
```

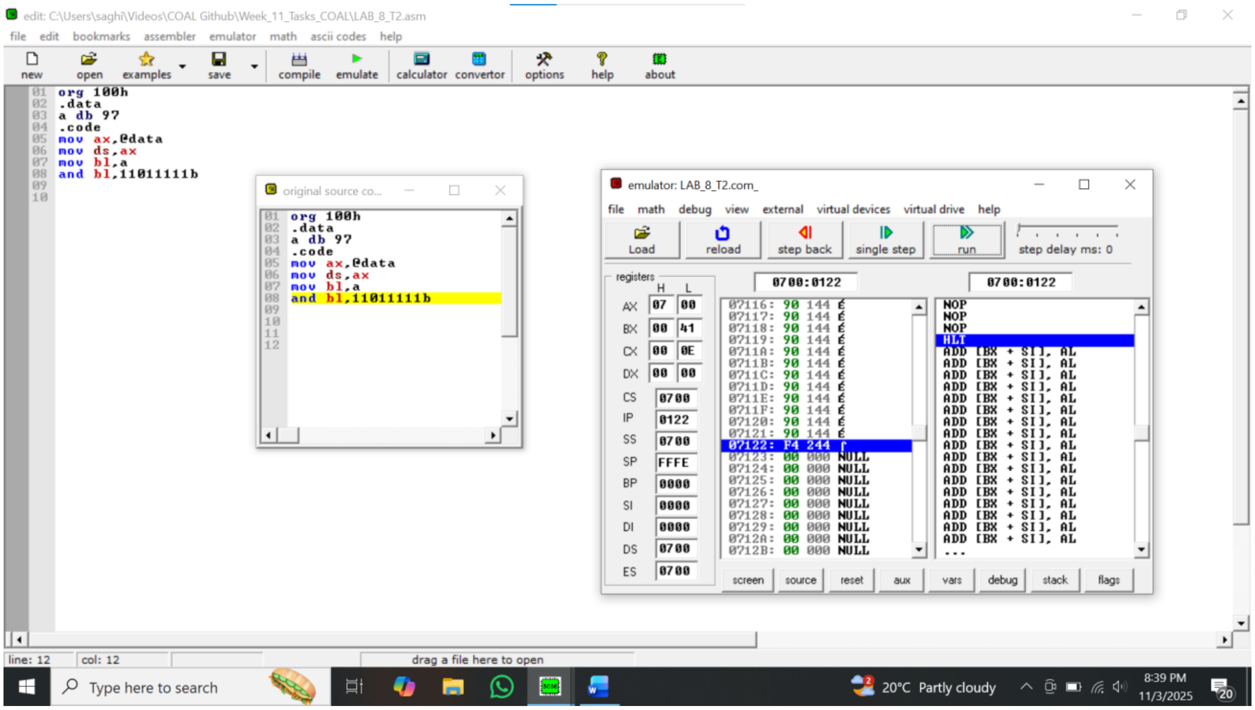
```
0700:0141                        0700:0141
0712D: 90 144 É                  NOP
0712E: 90 144 É                  NOP
0712F: 90 144 É                  NOP
07130: 90 144 É                  NOP
07131: 90 144 É                  NOP
07132: 90 144 É                  NOP
07133: 90 144 É                  NOP
07134: 90 144 É                  NOP
07135: 90 144 É                  NOP
07136: 90 144 É                  NOP
07137: 90 144 É                  NOP
07138: 90 144 É                  NOP
07139: 90 144 É                  NOP
0713A: 90 144 É                  NOP
0713B: 90 144 É                  NOP
0713C: 90 144 É                  NOP
0713D: 90 144 É                  NOP
0713E: 90 144 É                  NOP
0713F: 90 144 É                  NOP
07140: 90 144 É                  NOP
07141: F4 244 ⌠                  HLT
07142: 00 000 NULL               ADD [BX + SI], AL
                                 ...
```

Task 02 :

This program converts a lowercase letter to uppercase. It does this by using an AND operation with a specific bitmask to turn off the single bit that differentiates a lowercase letter from its uppercase counterpart.



```
org 100h
.data
a db 97
.code
mov ax,@data
mov ds,ax
mov bl,a
and bl,11011111b
```



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T2.asm

```
org 100h
.data
a db 97
.code
mov ax,@data
mov ds,ax
mov bl,a
and bl,11011111b
```

emulator: LAB_8_T2.com_

```
registers
     H   L
AX   07  00
BX   00  41
CX   00  0E
DX   00  00
CS   0700
IP   0122
SS   0700
SP   FFFE
BP   0000
SI   0000
DI   0000
DS   0700
ES   0700
```

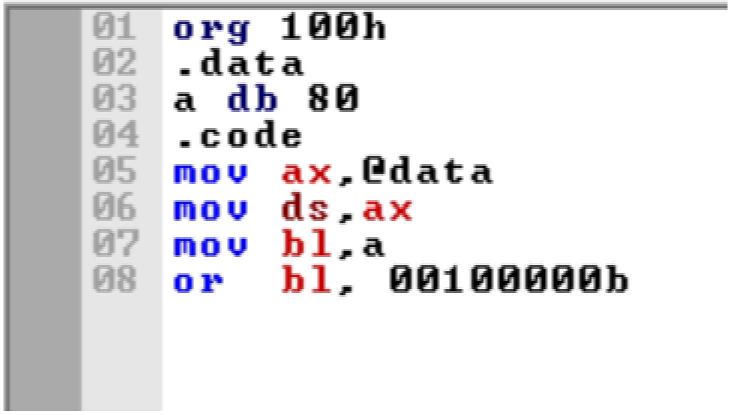
```
0700:0122
07116: 90 144 É
07117: 90 144 É
07118: 90 144 É
07119: 90 144 É
0711A: 90 144 É
0711B: 90 144 É
0711C: 90 144 É
0711D: 90 144 É
0711E: 90 144 É
0711F: 90 144 É
07120: 90 144 É
07121: 90 144 É
07122: F4 244 ſ
07123: 00 000 NULL
07124: 00 000 NULL
07125: 00 000 NULL
07126: 00 000 NULL
07127: 00 000 NULL
07128: 00 000 NULL
07129: 00 000 NULL
0712A: 00 000 NULL
0712B: 00 000 NULL
```

```
0700:0122
NOP
NOP
NOP
HLT
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
...
```

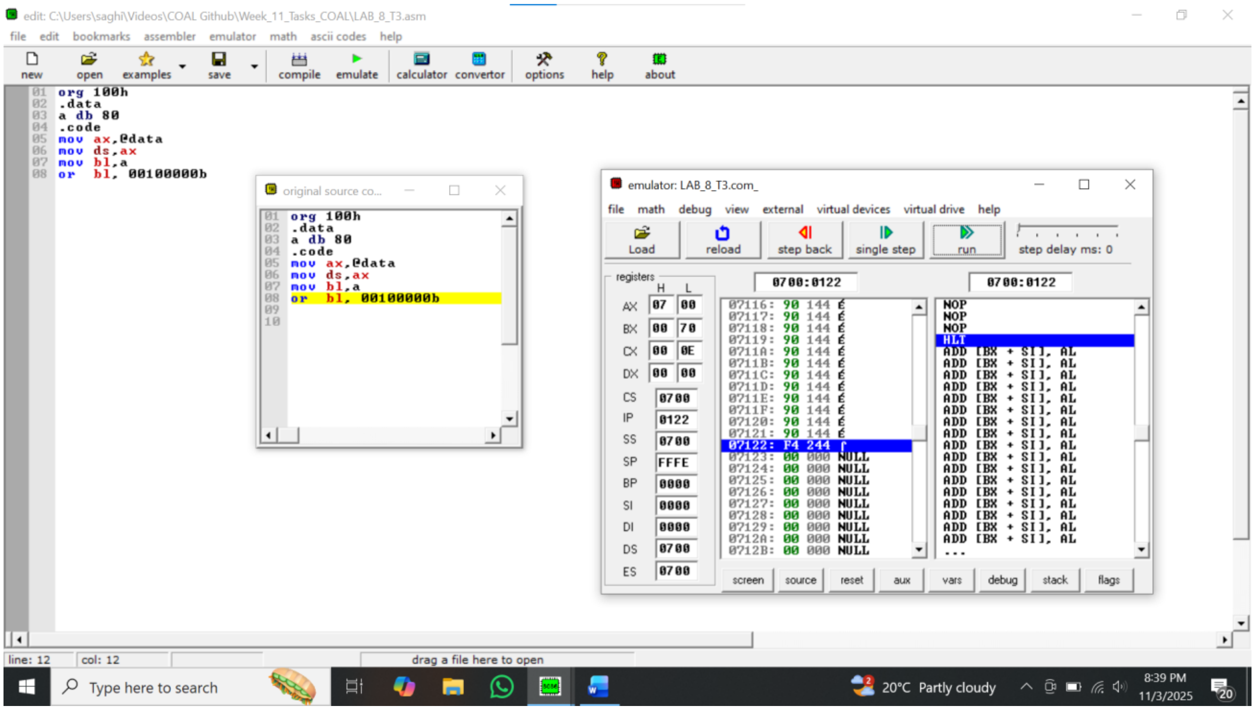
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
Load reload step back single step run step delay ms: 0
screen source reset aux vars debug stack flags
line: 12 col: 12 drag a file here to open
Type here to search
20°C Partly cloudy
8:39 PM
11/3/2025

Task 03 :

This is the opposite of T2; it converts an uppercase letter to lowercase. It uses an OR operation with a different bitmask to turn on the specific bit that makes a letter lowercase.



```
org 100h
.data
a db 80
.code
mov ax,@data
mov ds,ax
mov bl,a
or  bl, 00100000b
```



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T3.asm

```
org 100h
.data
a db 80
.code
mov ax,@data
mov ds,ax
mov bl,a
or bl, 00100000b
```

emulator: LAB_8_T3.com_

registers	H	L
AX	07	00
BX	00	70
CX	00	0E
DX	00	00
CS	0700	
IP	0122	
SS	0700	
SP	FFFE	
BP	0000	
SI	0000	
DI	0000	
DS	0700	
ES	0700	

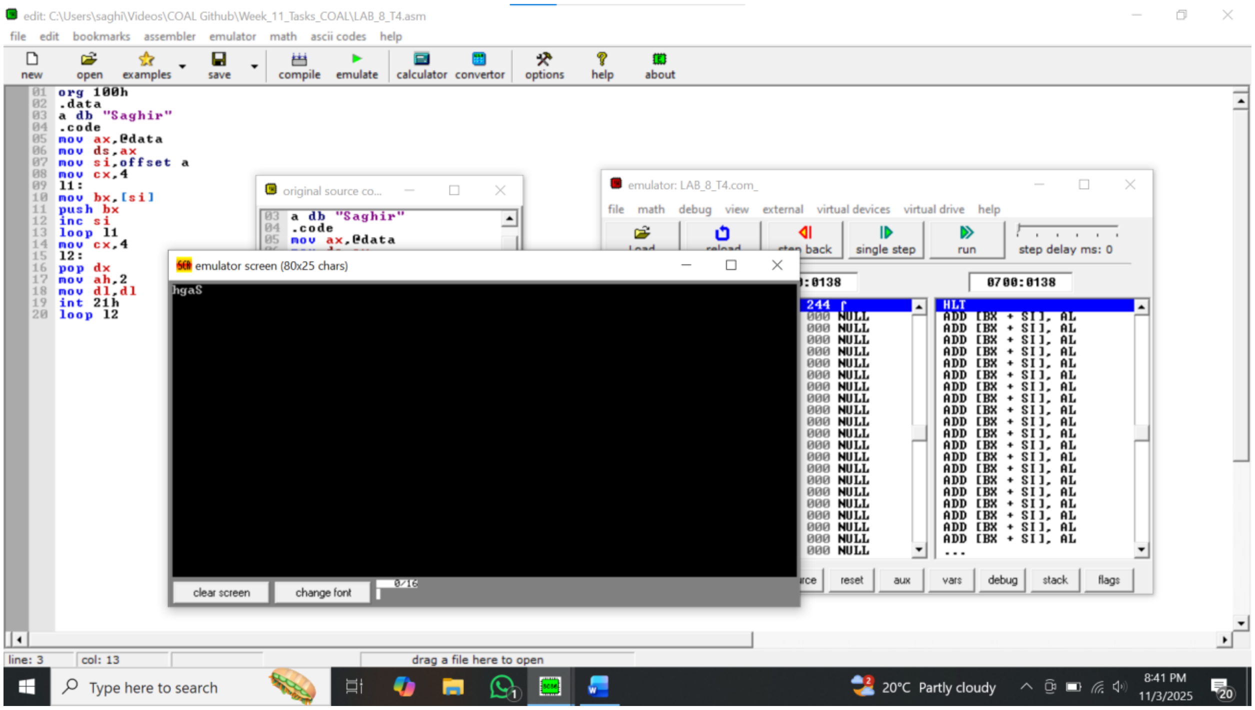
```
0700:0122

07116: 90 144 É
07117: 90 144 É
07118: 90 144 É
07119: 90 144 É
0711A: 90 144 É
0711B: 90 144 É
0711C: 90 144 É
0711D: 90 144 É
0711E: 90 144 É
0711F: 90 144 É
07120: 90 144 É
07121: 90 144 É
07122: F4 244 ⌠
07123: 00 000 NULL
07124: 00 000 NULL
07125: 00 000 NULL
07126: 00 000 NULL
07127: 00 000 NULL
07128: 00 000 NULL
07129: 00 000 NULL
0712A: 00 000 NULL
0712B: 00 000 NULL
```

```
0700:0122

NOP
NOP
NOP
HLT
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
...
```

file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
screen source reset aux vars debug stack flags
line: 12 col: 12 drag a file here to open
Type here to search
20°C Partly cloudy
8:39 PM
11/3/2025

Task 04:

It demonstrates a "Last-In, First-Out" stack. It takes the first four letters of a name, pushes them onto a stack in order, and then pops them off in reverse order, printing the name backwards.



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T4.asm

```
org 100h
.data
a db "Saghir"
.code
mov ax,@data
mov ds,ax
mov si,offset a
mov cx,4
l1:
mov bx,[si]
push bx
inc si
loop l1
mov cx,4
l2:
pop dx
mov ah,2
mov dl,dl
int 21h
loop l2
```

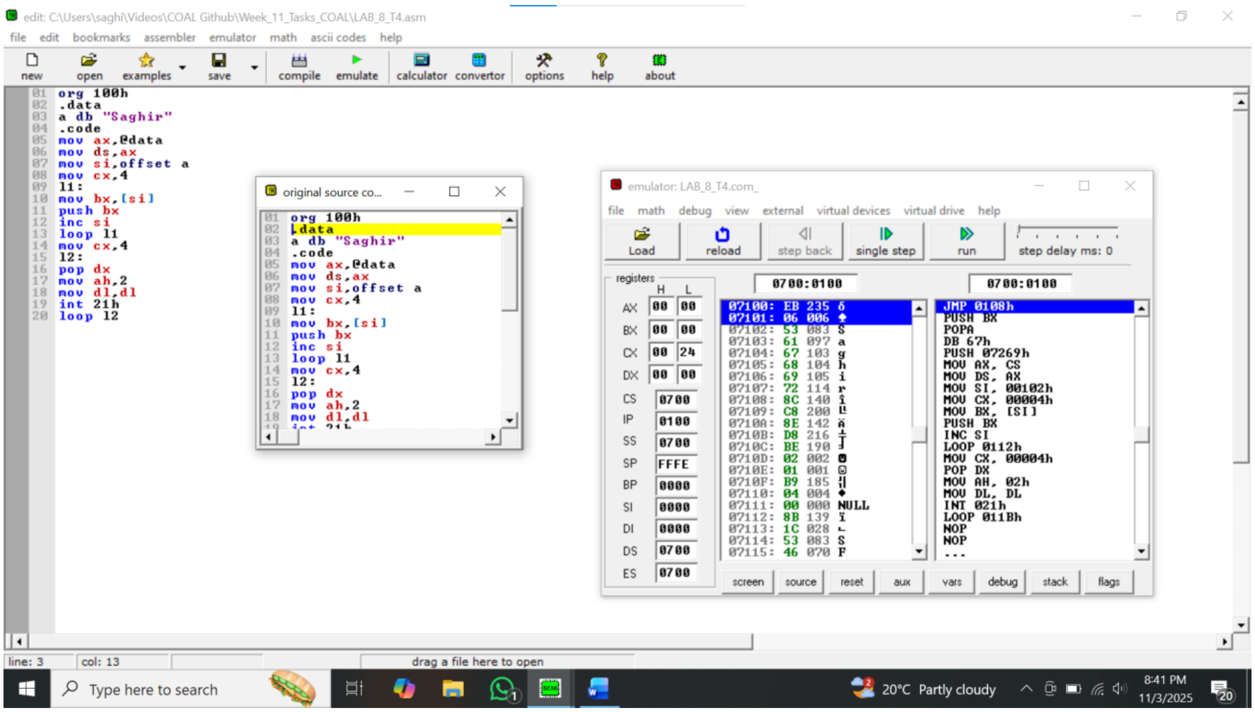
emulator screen (80x25 chars)

```
hgaS
```

emulator: LAB_8_T4.com_

```
0700:0138                    0700:0138
244 ſ                        HLT
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ADD [BX + SI], AL
000 NULL                     ...
```

file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
clear screen
change font
Load reload step back single step run
step delay ms: 0
reset aux vars debug stack flags
line: 3 col: 13
drag a file here to open
Type here to search
20°C Partly cloudy
8:41 PM
11/3/2025



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T4.asm

```
org 100h
.data
a db "Saghir"
.code
mov ax,@data
mov ds,ax
mov si,offset a
mov cx,4
l1:
mov bx,[si]
push bx
inc si
loop l1
mov cx,4
l2:
pop dx
mov ah,2
mov dl,dl
int 21h
loop l2
```

emulator: LAB_8_T4.com_

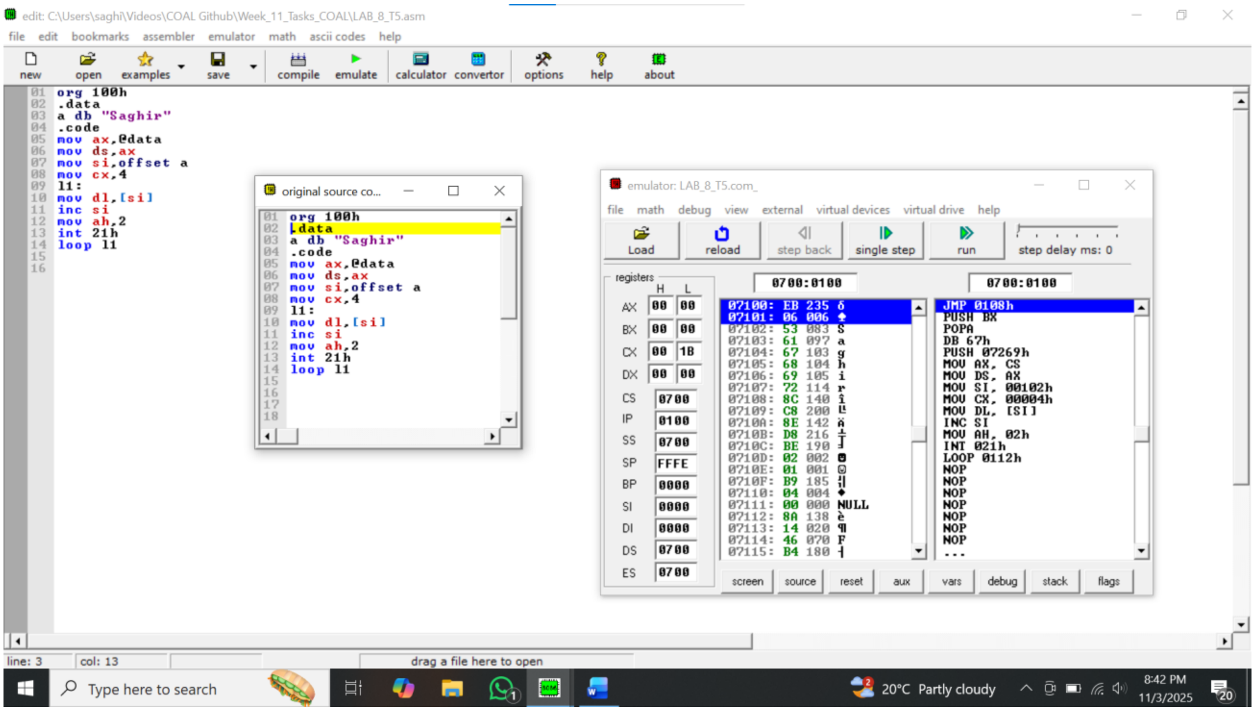
registers	H	L
AX	00	00
BX	00	00
CX	00	24
DX	00	00
CS	0700	
IP	0100	
SS	0700	
SP	FFFE	
BP	0000	
SI	0000	
DI	0000	
DS	0700	
ES	0700	

```
0700:0100                    0700:0100
07100: EB 235 δ              JMP 0108h
07101: 06 006 ♠              PUSH BX
07102: 53 083 S              POPA
07103: 61 097 a              DB 67h
07104: 67 103 g              PUSH 07269h
07105: 68 104 h              MOV AX, CS
07106: 69 105 i              MOV DS, AX
07107: 72 114 r              MOV SI, 00102h
07108: 8C 140 î              MOV CX, 00004h
07109: C8 200 ╚              MOV BX, [SI]
0710A: 8E 142 Ä              PUSH BX
0710B: D8 216 ╪              INC SI
0710C: BE 190 ╛              LOOP 0112h
0710D: 02 002 ☻              MOV CX, 00004h
0710E: 01 001 ☺              POP DX
0710F: B9 185 ╣              MOV AH, 02h
07110: 04 004 ♦              MOV DL, DL
07111: 00 000 NULL           INT 021h
07112: 8B 139 ï              LOOP 011Bh
07113: 1C 028 ∟              NOP
07114: 53 083 S              NOP
07115: 46 070 F              ...
```

file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
Load reload step back single step run
step delay ms: 0
screen source reset aux vars debug stack flags
line: 3 col: 13
drag a file here to open
Type here to search
20°C Partly cloudy
8:41 PM
11/3/2025

Task 05 :

This program prints a string in the correct order. It reads each character from memory one by one and immediately sends it to the screen, displaying the first four letters of the name as stored.



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T5.asm

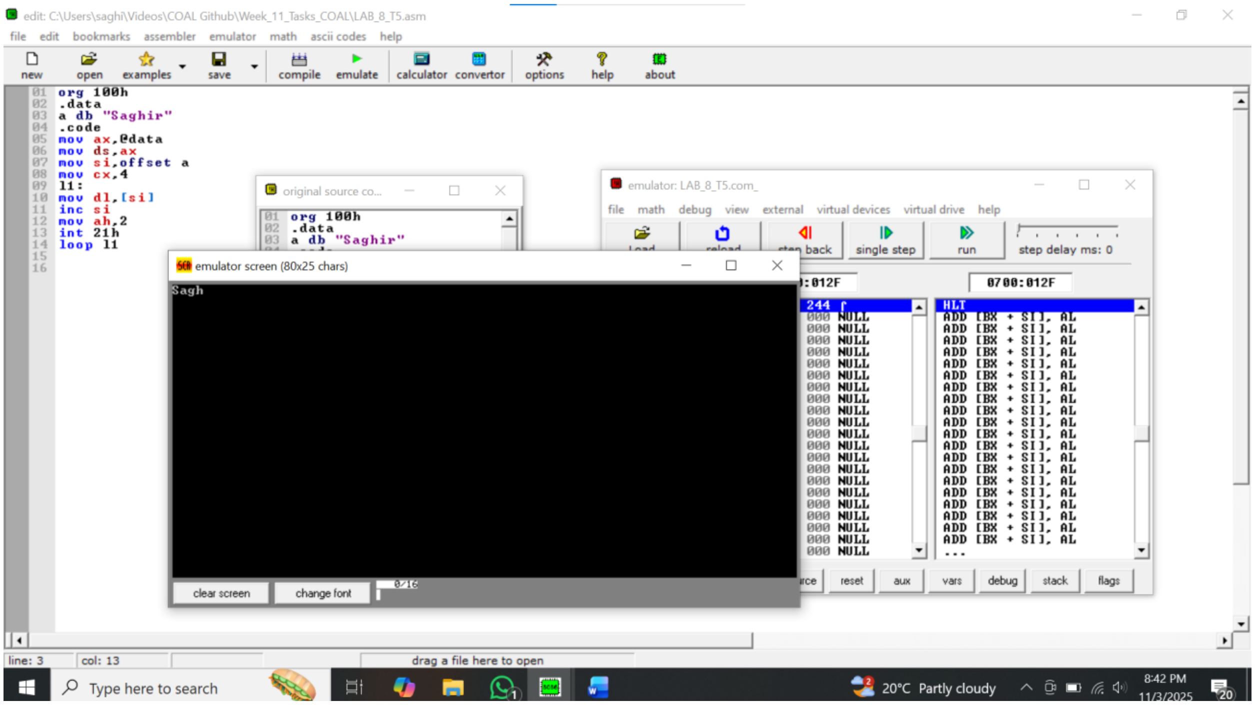
```
org 100h
.data
a db "Saghir"
.code
mov ax,@data
mov ds,ax
mov si,offset a
mov cx,4
l1:
mov dl,[si]
inc si
mov ah,2
int 21h
loop l1
```

emulator: LAB_8_T5.com_

```
registers  H   L
AX        00  00
BX        00  00
CX        00  1B
DX        00  00
CS  0700
IP  0100
SS  0700
SP  FFFE
BP  0000
SI  0000
DI  0000
DS  0700
ES  0700
```

```
0700:0100                  0700:0100
07100: EB 235 δ            JMP 0108h
07101: 06 006 ♠            PUSH BX
07102: 53 083 S            POPA
07103: 61 097 a            DB 67h
07104: 67 103 g            PUSH 07269h
07105: 68 104 h            MOV AX, CS
07106: 69 105 i            MOV DS, AX
07107: 72 114 r            MOV SI, 00102h
07108: 8C 140 î            MOV CX, 00004h
07109: C8 200 ╚            MOV DL, [SI]
0710A: 8E 142 Ä            INC SI
0710B: D8 216 ╪            MOV AH, 02h
0710C: BE 190 ╛            INT 021h
0710D: 02 002 ☻            LOOP 0112h
0710E: 01 001 ☺            NOP
0710F: B9 185 ╣            NOP
07110: 04 004 ♦            NOP
07111: 00 000 NULL         NOP
07112: 8A 138 è            NOP
07113: 14 020 ¶            NOP
07114: 46 070 F            NOP
07115: B4 180 ┤            ...
```

file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
Load reload step back single step run step delay ms: 0
screen source reset aux vars debug stack flags
line: 3 col: 13 drag a file here to open
Type here to search
20°C Partly cloudy
8:42 PM
11/3/2025



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T5.asm

```
org 100h
.data
a db "Saghir"
.code
mov ax,@data
mov ds,ax
mov si,offset a
mov cx,4
l1:
mov dl,[si]
inc si
mov ah,2
int 21h
loop l1
```

emulator screen (80x25 chars)

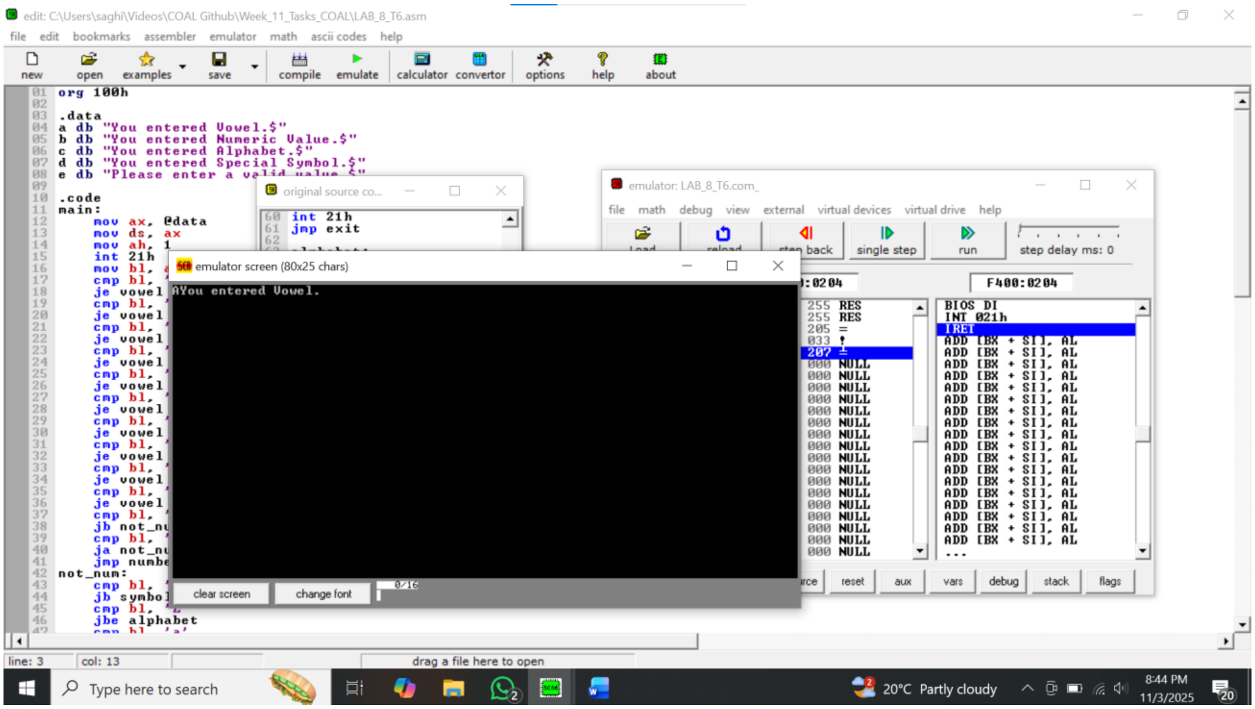
```
Sagh
```

```
0700:012F
HLT
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
...
```

file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
emulator: LAB_8_T5.com_
clear screen change font
step back single step run step delay ms: 0
reset aux vars debug stack flags
line: 3 col: 13 drag a file here to open
Type here to search
20°C Partly cloudy
8:42 PM
11/3/2025

Task 06 :

This is an interactive character classifier. It takes a single user input, checks if it's a vowel, a number, an alphabet letter, or a symbol, and then prints the corresponding message based on what it finds.



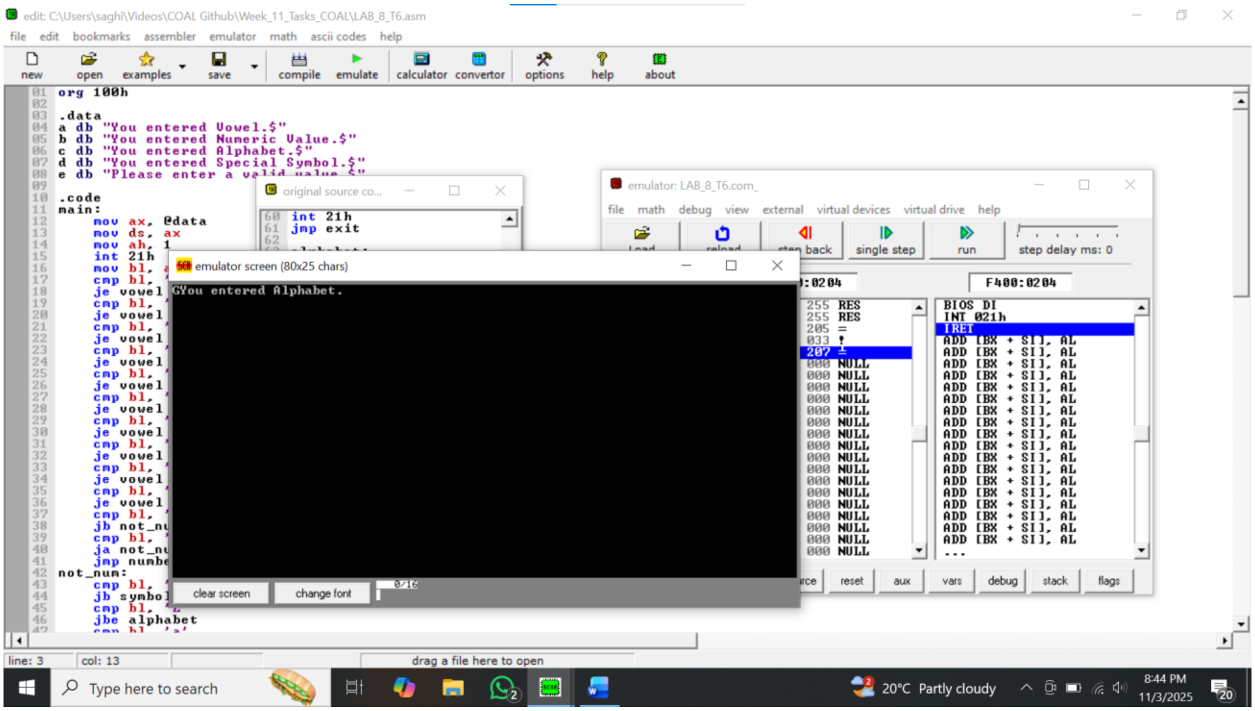
edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T6.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
emulator screen (80x25 chars)
clear screen change font
emulator: LAB_8_T6.com_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
rce reset aux vars debug stack flags
line: 3 col: 13 drag a file here to open
Type here to search
20°C Partly cloudy
8:44 PM
11/3/2025

```
org 100h

.data
a db "You entered Vowel.$"
b db "You entered Numeric Value.$"
c db "You entered Alphabet.$"
d db "You entered Special Symbol.$"
e db "Please enter a valid value.$"

.code
main:
      mov ax, @data
      mov ds, ax
      mov ah, 1
      int 21h
      mov bl, a
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      jb not_nu
      cmp bl,
      ja not_nu
      jmp numbe
not_num:
      cmp bl,
      jb symbol
      cmp bl, 'z
      jbe alphabet
```

```
int 21h
jmp exit
```

```
AYou entered Vowel.
```

```
:0204            F400:0204
255 RES          BIOS DI
255 RES          INT 021h
205 =            IRET
033 !            ADD [BX + SI], AL
207 ╧            ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ...
```



edit: C:\Users\saghi\Videos\COAL Github\Week_11_Tasks_COAL\LAB_8_T6.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source co...
emulator screen (80x25 chars)
clear screen change font
emulator: LAB_8_T6.com_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
rce reset aux vars debug stack flags
line: 3 col: 13 drag a file here to open
Type here to search
20°C Partly cloudy
8:44 PM
11/3/2025

```
org 100h

.data
a db "You entered Vowel.$"
b db "You entered Numeric Value.$"
c db "You entered Alphabet.$"
d db "You entered Special Symbol.$"
e db "Please enter a valid value.$"

.code
main:
      mov ax, @data
      mov ds, ax
      mov ah, 1
      int 21h
      mov bl, a
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      je vowel
      cmp bl,
      jb not_nu
      cmp bl,
      ja not_nu
      jmp numbe
not_num:
      cmp bl,
      jb symbol
      cmp bl, 'z
      jbe alphabet
```

```
int 21h
jmp exit
```

```
GYou entered Alphabet.
```

```
:0204            F400:0204
255 RES          BIOS DI
255 RES          INT 021h
205 =            IRET
033 !            ADD [BX + SI], AL
207 ╧            ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ADD [BX + SI], AL
000 NULL         ...
```