CSE 331L / EEE 332L: (Lab 8),

Section: 7 & 8, Fall 2019

Topic:

1. Library: emu8086.inc

To use any of the functions in emu8086.inc you should have the following line in the beginning of your source file:

include 'emu8086.inc'

emu8086.inc defines the following **macros**:

- **PUTC char**: macro with 1 parameter, prints out an ASCII char at current cursor position.
- **GOTOXY col, row**: macro with 2 parameters, sets cursor position.
- **PRINT string**: macro with 1 parameter, prints out a string.
- **PRINTN string**: macro with 1 parameter, prints out a string. The same as PRINT but automatically adds "carriage return" at the end of the string.
- **CURSOROFF**: turns off the text cursor.
- **CURSORON**: turns on the text cursor.

```
include emu8086.inc

ORG    100h

PRINT n'Hello World!'
PUTC 'B'

GOTOXY 10, 5

PUTC 65         ; 65 - is an ASCII code for 'A'
PUTC 'B'

RET             ; return to operating system.
END             ; directive to stop the compiler.
```

emu8086.inc also defines the following procedures:

- **PRINT_STRING** - procedure to print a null terminated string at current cursor position, receives address of string in DS:SI register. To use it declare: DEFINE_PRINT_STRING before END directive.
- **PTHIS** - procedure to print a null terminated string at current cursor position (just as PRINT_STRING), but receives address of string from Stack. The ZERO TERMINATED string should be defined just after the CALL instruction. For example:

  ```
  CALL PTHIS
  db 'Hello World!', 0
  ```

  To use it declare: DEFINE_PTHIS before END directive.
- **GET_STRING** - procedure to get a null terminated string from a user, the received string is written to buffer at DS:DI, buffer size should be in DX. Procedure stops the input when 'Enter' is pressed. To use it declare: DEFINE_GET_STRING before END directive.
- **CLEAR_SCREEN** - procedure to clear the screen, (done by scrolling entire screen window), and set cursor position to top of it. To use it declare: DEFINE_CLEAR_SCREEN before END directive.
- **SCAN_NUM** - procedure that gets the multi-digit SIGNED number from the keyboard, and stores the result in CX register. To use it declare: DEFINE_SCAN_NUM before END directive.
- **PRINT_NUM** - procedure that prints a signed number in AX register. To use it declare: DEFINE_PRINT_NUM and DEFINE_PRINT_NUM_UNS before END directive.
- **PRINT_NUM_UNS** - procedure that prints out an unsigned number in AX register. To use it declare: DEFINE_PRINT_NUM_UNS before END directive.

To use any of the above procedures you should first declare the function in the bottom of your file (but before the **END** directive), and then use **CALL** instruction followed by a procedure name. For example:

Example: take a number as input and display it.

```
include 'emu8086.inc'

ORG    100h
.DATA
      msg1   DB  'Enter the number: ', 0
.CODE

LEA    SI, msg1         ; ask for the number
CALL   print_string   ; print the msg
CALL   scan_num     ; get number in CX.

MOV    AX, CX        ; copy the number to AX.

; print the following string:
CALL   pthis
DB  13, 10, 'You have entered: ', 0

CALL   print_num      ; print number in AX.

RET                ; return to operating system.


DEFINE_SCAN_NUM
DEFINE_PRINT_STRING
DEFINE_PRINT_NUM
DEFINE_PRINT_NUM_UNS  ; required for print_num.
DEFINE_PTHIS

END                ; directive to stop the compiler.
```

Example: take two numbers as input and display their summation.

```
include 'emu8086.inc'

ORG    100h

.data
        msg1   DB  'Enter a number: ', 0
        msg2   DB  'Enter another number: ', 0
.code

LEA     SI, msg1        ; ask for the number
CALL   print_string   ;
CALL   scan_num      ; get number in CX.

MOV   AX, CX          ; copy the number to AX.

CALL pthis
DB 13, 10, 0

LEA     SI, msg2        ; ask for the number
CALL   print_string   ;
CALL   scan_num

MOV BX, CX
ADD AX, BX

CALL   pthis
DB  13, 10, 'THE SUM IS: ', 0

CALL   print_num     ; print number in AX.
RET             ; return to operating system.


DEFINE_SCAN_NUM
DEFINE_PRINT_STRING
DEFINE_PRINT_NUM
DEFINE_PRINT_NUM_UNS  ; required for print_num.
DEFINE_PTHIS
END             ; directive to stop the compiler.
```