Lab: 8
Familiarization with BIOS Service INT 10H in Assembly Language Programming

BIOS Service
BIOS provides interrupt service 10H for video display control. `INT 10H` also has many functions like `INT 21H`, and some of them are given below. The function no of the service is to be loaded in register AH and the other registers are loaded with the data as required by the function before interrupt call.

BIOS Service `INT 10H`

Func. No.	Description
00H	Set video mode. Load the required mode in `AL`. This operation also clears the screen.
01H	Set cursor size. To set cursor vertically set the register `CX` as: CH (bits 4-0): starting scan line CL (bits 4-0): ending scan line
02H	Set cursor position anywhere on a screen according to row:column coordinates. Set the registers as follows: BH: page number (0 is the default), DH: row, and DL: column.
03H	Return cursor status i.e. to determine the present row, column and size to the cursor. Store the page number in BH. The operation leaves `AX` and `BX` unchanged and returns these values: CH: Starting scan line CL: Ending scan line DH: Row DL: Column
05H	Select the page that is to be displayed. We can create different pages and request alternating between pages.
06H	Scroll upward of lines in a specified area of the screen. Displayed lines scroll off at the top and blank lines appear at the bottom. Setting AL to 0 caused the entire screen to scroll up, effectively clearing it. Setting a nonzero value in AL causes the number of lines to scroll up. Set the following registers as: AL: Number of rows (00 for full screen) CX: Starting row, Column BH: Attribute of pixel value DX: Ending row, Column
07H	Scroll down screen. Scrolling down the screen causes the bottom lines to scroll off and blank lines to appear at the top. It works the same as function 06H, except the fact that this operation scrolls down. Set the following registers as: AL: Number of rows (00 for full screen) CX: Starting row, Column BH: Attribute or pixel value DX: Ending row, Column
08H	Read character and its attribute at cursor from the video display area. Before calling interrupt, set the page number in BH register.
09H	Display a specified number of characters at cursor according to given attribute. Set the registers as: AL: ASCII character BL: Attribute or pixel value BH: Page number CX: Count The count in CX specifies the number of times the operation is to repetitively display the character in AL.
0AH	Display character at cursor. The difference with 09H is that function 09H sets the attribute whereas function 0AH used the current value. AL: ASCII character BL: Pixel value (graphics mode only) BH: Page number CX: Count
0BH	Set the color palette. The value in BH (00 or 01) determines the purpose of BL BH = 00: Select the background color, where BL contains the color value in bits 0-3 (any of 16 colors). BH = 01: Select the palette for graphics, where BL contains the palette (0 or 1).
0CH	Display a selected color (background and palette) in graphics mode. Set the registers as: AL: Color of the pixel CX: Column BH: Page number DX: Row
0DH	Read pixel dot to determine its color value. For this set page number in BH, column in CX, and row in DX. The operation returns the pixel color in AL.
0EH	Monitor is used as a terminal for simple displays in test and graphics modes. For this set AL by the character to display, and BL by the foreground color.
0FH	Get current video mode. The operation returns the values as: AL: Current video mode AH: umber of screen columns BH: Active video page

Assignments:

1. Write an assembly language program to scroll a window from row 5, column 20 to row 20, column 60 with a reverse video attributes. Then locate the cursor at row 12, column 30. And display a string as "Programming in Assembly Language is Fun".

2. Write an assembly language program that takes a string (having 60 characters at max.) as input from user, and display the string at the center of the clear screen.

3. Write an assembly language program that takes a string (having 24 characters at max.) from the user and display each character at the center of each line.

4. Write an assembly language program that takes a string (having 14 characters at max.) input from user and scroll a window of size 20×20 at the center of screen. Then display the string at the center of scrolled window. (You can choose the color attribute yourself).

5. Write a program that a string from the user and display each word in new line diagonally from upper left towards bottom right in a clear screen. If the string is "Programming in Assembly Language is Fun", it shoud be displayed as
Programming
       in
          Assembly
               Language
                  is
                    Fun