Laboratory Exercise No.:	3	**Date Performed:**	
Laboratory Exercise Title:	Assembly Language Instructions		
Name of Student(s):	Christian Jay Y. Gallardo Jhon Fil Tizon	**Document Version:**	1

Laboratory Report

Activity #34-1

```
; 34-1  (optimized)

.model small
.stack 100h

.data
; --- UI text ---
menuTitle   db 13,10,'Menu$'
menuOpt1    db 13,10,'1 - Horizontal Stripes$'
menuOpt2    db 13,10,'2 - Vertical Stripes$'
menuOpt3    db 13,10,'3 - Exit$'
promptTimes db 13,10,'Press any key to continue.$'

; --- input buffer for DOS AH=0Ah ---
; [0]=max count, [1]=actual len, [2..]=data
kbdBuf      db 3,0, 3 dup(0)

.code
MAIN PROC
    mov ax, @data
    mov ds, ax

;================ MENU LOOP ================
ShowMenu:
```



```
    ; clear with blue bg + yellow fg (BH carries attribute for 06h)
    mov bh, 1Eh
    call ClearScreen

    lea dx, menuTitle   ; display menu
    call PrintStr
    lea dx, menuOpt1
    call PrintStr
    lea dx, menuOpt2
    call PrintStr
    lea dx, menuOpt3
    call PrintStr

    ; read one key (DOS buffered line, we only take first)
    mov ah, 0Ah
    lea dx, kbdBuf
    int 21h

    mov al, kbdBuf+2
    cmp al, '1'
    je  DoHorizontal
    cmp al, '2'
    je  DoVertical
    cmp al, '3'
    je  ExitProgram
    jmp ShowMenu

;================ HORIZONTAL ================
; 4 bands x 5 rows each, attributes: 10h,20h,40h,50h
DoHorizontal:
    ; band 0: rows 0..4  (5 rows)  attr 10h
    mov bl, 10h
    mov dh, 0
    mov si, 5
HBandLoop0:
    call FillRow80
```

```
    inc dh
    dec si
    jnz HBandLoop0

    ; band 1: rows 5..9  attr 20h
    mov bl, 20h
    mov si, 5
HBandLoop1:
    call FillRow80
    inc dh
    dec si
    jnz HBandLoop1

    ; band 2: rows 10..14 attr 40h
    mov bl, 40h
    mov si, 5
HBandLoop2:
    call FillRow80
    inc dh
    dec si
    jnz HBandLoop2

    ; band 3: rows 15..19 attr 50h
    mov bl, 50h
    mov si, 5
HBandLoop3:
    call FillRow80
    inc dh
    dec si
    jnz HBandLoop3

    lea dx, promptTimes
    call PrintStr
    mov ah, 00h         ; wait key
    int 16h
    jmp ShowMenu
```

```
;================ VERTICAL ================
; 4 bands x 20 columns each, attributes: E0h,20h,40h,50h
DoVertical:
    ; clear first to default (grey on black)
    mov bh, 07h
    call ClearScreen

    ; band 0: cols 0..19  attr E0h
    mov bl, 0E0h
    mov dl, 0
    mov cl, 20
VBand0:
    call FillCol25
    inc dl
    dec cl
    jnz VBand0

    ; band 1: cols 20..39  attr 20h
    mov bl, 20h
    mov cl, 20
VBand1:
    call FillCol25
    inc dl
    dec cl
    jnz VBand1

    ; band 2: cols 40..59  attr 40h
    mov bl, 40h
    mov cl, 20
VBand2:
    call FillCol25
    inc dl
    dec cl
    jnz VBand2
```

```
    ; band 3: cols 60..79  attr 50h
    mov bl, 50h
    mov cl, 20
VBand3:
    call FillCol25
    inc dl
    dec cl
    jnz VBand3

    lea dx, promptTimes
    call PrintStr
    mov ah, 00h         ; wait key
    int 16h
    jmp ShowMenu

;================ PROCS =================

; Clear text screen (80x25)
; IN:  BH = attribute for blanking
ClearScreen PROC
    mov ax, 0600h       ; scroll up full window
    mov cx, 0000h       ; upper-left (row 0, col 0)
    mov dx, 184Fh       ; lower-right (row 24, col 79)
    int 10h
    ret
ClearScreen ENDP

; Print $-terminated string at DS:DX via DOS
PrintStr PROC
    mov ah, 09h
    int 21h
    ret
PrintStr ENDP

; Fill one full row (80 characters) with attribute BL at row DH
```

```
; Uses page 0, character ' '.
FillRow80 PROC
    push ax
    push bx
    push cx
    push dx

    mov dl, 0          ; col 0
    mov ah, 02h        ; set cursor
    mov bh, 0
    int 10h

    mov ah, 09h        ; write character/attribute
    mov al, ' '
    mov bh, 0
    mov cx, 80         ; THIS clobbers CX; preserved by pushes
    ; BL already holds attribute
    int 10h

    pop dx
    pop cx
    pop bx
    pop ax
    ret
FillRow80 ENDP


; Fill one column (25 rows) with attribute BL at column DL
; Writes single space per row.
FillCol25 PROC
    push dx
    mov dh, 0          ; start row 0
    mov si, 25
FillCol_Loop:
    mov ah, 02h        ; set cursor to (DH, DL)
    mov bh, 0
```

```
    int 10h

    mov ah, 09h         ; write one char with BL attribute
    mov al, ' '
    mov bh, 0
    mov cx, 1
    int 10h

    inc dh
    dec si
    jnz FillCol_Loop
    pop dx
    ret
FillCol25 ENDP

;================ EXIT =================
ExitProgram:
    mov ah, 4Ch
    int 21h
END MAIN
```

Activity #34-2

```
;34-2

org 100h

.data
prompt  db 13,10,'Enter the text "This will be displayed on the screen.": $'
buffer  db 50, 0, 50 dup('$')     ; DOS 0Ah input buffer
msg     db 13,10,'Now displaying...$'

.code
main proc
   mov ax, @data
   mov ds, ax

   ; Ask for input
   lea dx, prompt
   mov ah, 09h
   int 21h

   ; Read string into buffer
   mov ah, 0Ah
   lea dx, buffer
   int 21h

   ; Show message
   lea dx, msg
   mov ah, 09h
   int 21h

   ; ==============================
   ; Display loop from row 0 to 24
   ; ==============================
   mov dh, 0         ; start row
   mov dl, 39        ; start col
DisplayLoop:
   mov ah, 02h       ; set cursor
   mov bh, 0
   int 10h

   mov si, offset buffer+2   ; string begins at buffer+2
print_chars:
   mov al, [si]
   cmp al, 0Dh               ; stop at Enter
   je  end_line
   cmp al, 0Ah
   je  end_line
   cmp al, 0
   je  end_line
```

```
    mov ah, 09h
    mov bh, 0
    mov cx, 1
    mov bl, 06h             ; brown text on black
    int 10h

    inc si
    jmp print_chars

end_line:
    inc dh
    cmp dh, 25              ; 25 rows max
    jb  DisplayLoop

    ; Exit
    mov ah, 4Ch
    int 21h
main endp
end main
```