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 #35-1

```
; 35-1 (optimized)

.model small
.stack 100h

.data
menuTitle   db 13,10,'Menu$'
menu1       db 13,10,'1 - Horizontal Stripes$'
menu2       db 13,10,'2 - Vertical Stripes$'
menu3       db 13,10,'3 - Checkerboard$'
menu4       db 13,10,'4 - Exit$'
contMsg     db 13,10,'Press any key to continue.$'

; attributes
hAttrs      db 10h,20h,40h,50h          ; 4 bands (5 rows each)
vAttrs      db 0E0h,20h,40h,50h          ; 4 bands (20 cols each)

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

; ===== Menu =====
ShowMenu:
```



```
    mov bh, 1Eh
    call ClearScreen

    lea dx, menuTitle
    call PrintStr
    lea dx, menu1
    call PrintStr
    lea dx, menu2
    call PrintStr
    lea dx, menu3
    call PrintStr
    lea dx, menu4
    call PrintStr

    ; wait single key
    mov ah, 01h
    int 21h           ; AL = ASCII

    cmp al, '1'
    je  DoHorizontal

    cmp al, '2'
    je  DoVertical

    cmp al, '3'
    je  DoChecker

    cmp al, '4'
    je  ExitProgram

    jmp ShowMenu


; ===== Horizontal: 4 bands x 5 rows =====
DoHorizontal:
    xor dh, dh        ; row = 0
```

```
    lea si, hAttrs
    mov di, 4          ; 4 bands
HBandLoop:
    mov bl, [si]
    mov cl, 5          ; 5 rows per band
HRowLoop:
    call FillRow80     ; uses BL attr at row DH
    inc dh
    dec cl
    jnz HRowLoop
    inc si
    dec di
    jnz HBandLoop

    lea dx, contMsg
    call PrintStr
    mov ah, 00h
    int 16h
    jmp ShowMenu

; ===== Vertical: 4 bands x 20 cols =====
DoVertical:
    mov bh, 07h
    call ClearScreen

    xor dl, dl         ; col = 0
    lea si, vAttrs
    mov di, 4          ; 4 bands
VBandLoop:
    mov bl, [si]
    mov cx, 20         ; 20 columns per band
VColLoop:
    call FillCol25     ; uses BL attr at col DL
    inc dl
    loop VColLoop
    inc si
```

```
    dec di
    jnz VBandLoop

    lea dx, contMsg
    call PrintStr
    mov ah, 00h
    int 16h
    jmp ShowMenu

; ===== Checkerboard (80x25) =====
DoChecker:
    mov bh, 07h
    call ClearScreen

    xor dh, dh          ; row
RowChk:
    xor dl, dl          ; col
ColChk:
    ; set cursor
    mov ah, 02h
    mov bh, 0
    int 10h

    ; attr based on (row+col)&1
    mov bl, dh
    add bl, dl
    and bl, 1
    jz  EvenCell
    mov bl, 1Fh         ; blue fg / white bg
    jmp PrintCell
EvenCell:
    mov bl, 70h         ; black fg / white bg
PrintCell:
    ; write one space with attribute
    mov ah, 09h
    mov al, ' '
```

```
    mov bh, 0
    mov cx, 1
    int 10h

    inc dl
    cmp dl, 80
    jb  ColChk

    inc dh
    cmp dh, 25
    jb  RowChk

    lea dx, contMsg
    call PrintStr
    mov ah, 00h
    int 16h
    jmp ShowMenu

; ===== Procs =====
; Clear 80x25 with attribute in BH
ClearScreen PROC
    push ax
    push cx
    push dx
    mov ax, 0600h
    mov cx, 0000h
    mov dx, 184Fh
    int 10h
    pop dx
    pop cx
    pop ax
    ret
ClearScreen ENDP

; Print $-terminated string at DS:DX
PrintStr PROC
```

```
    push ax
    mov ah, 09h
    int 21h
    pop ax
    ret
PrintStr ENDP

; Fill current row DH with 80 spaces, attr BL
FillRow80 PROC
    push ax
    push bx
    push cx
    push dx
    mov dl, 0
    mov ah, 02h
    mov bh, 0
    int 10h

    mov ah, 09h
    mov al, ' '
    mov bh, 0
    mov cx, 80
    ; BL already set
    int 10h
    pop dx
    pop cx
    pop bx
    pop ax
    ret
FillRow80 ENDP

; Fill column DL across 25 rows with spaces, attr BL
FillCol25 PROC
    push ax
    push bx
    push cx
```

```
    push dx
    mov dh, 0
    mov si, 25
FillColLoop:
    mov ah, 02h
    mov bh, 0
    int 10h
    mov ah, 09h
    mov al, ' '
    mov bh, 0
    mov cx, 1
    int 10h
    inc dh
    dec si
    jnz FillColLoop
    pop dx
    pop cx
    pop bx
    pop ax
    ret
FillCol25 ENDP

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