Program Code:

;Reverse Printing a Name Using String Instruction (MOVSB) In Assembly Language

```
.model small
.stack 100h
.data
Name1 db 'HEFAR LUDBA$'
Reversename2 db 11 dup('0$')        ;dup operator use for declaring a
large array
.code

start:

mov ax,@data                              ;initialize Data
Segment
mov ds,ax
mov es,ax

;Graphics Work Background Color Change
;Medium Resolution CGA
```

```asm
Mov ah, 0            ;set Graphics mode
mov al,04h   ;Medium Resolution mode 4
int 10h                      ;cursor position


mov ah,0bh           ;function for background
mov bh,00h           ;select background color
mov bl,6             ;Brown 0 Palette
int 10h


mov ah,02            ;set cursor position
mov bh,0             ;for page
mov dh,5     ;for row
mov dl,15    ;for column
int 10h


;-------------------------------------


lea si,Name1+10
lea di,Reversename2


std                                          ;direction flag
for reverse printing
```

```asm
mov cx,11                              ;Set count register to
11 Because in my name Alphabet is 11


MoveReverse:

movsb                                  ;String instruction
DS:SI into ES:DI (movsb) Move string from one variable to another
variable

add di,2                               ;Destination Index


loop MoveReverse


mov dx, offset Reversename2            ;Print string

mov ah,09

int 21h


;-----LINE Down-----

Mov ah,0ch                             ;Pixel function write

Mov al,2                               ;Pixel color Magenta
1 palette

Mov cx,115                             ;column

mov dx,50                              ;row


LD:

int 10h
```

```asm
inc  cx                                      ;to next column
increment

cmp cx,210                                   ;complete

jle LD


;-----LINE Up-----

Mov ah,0ch

Mov al,2

Mov cx,115

mov dx,35


LU:

int 10h

inc  cx

cmp cx,210

jle LU


Mov ah,0                                     ;read keyboard

int 16h                                      ;int 16h will
communicate with keyboard


Mov ax,3                                     ;set text mode
```

int 10h

mov ah, 4ch   ;exit

int 21h

end start

Output: