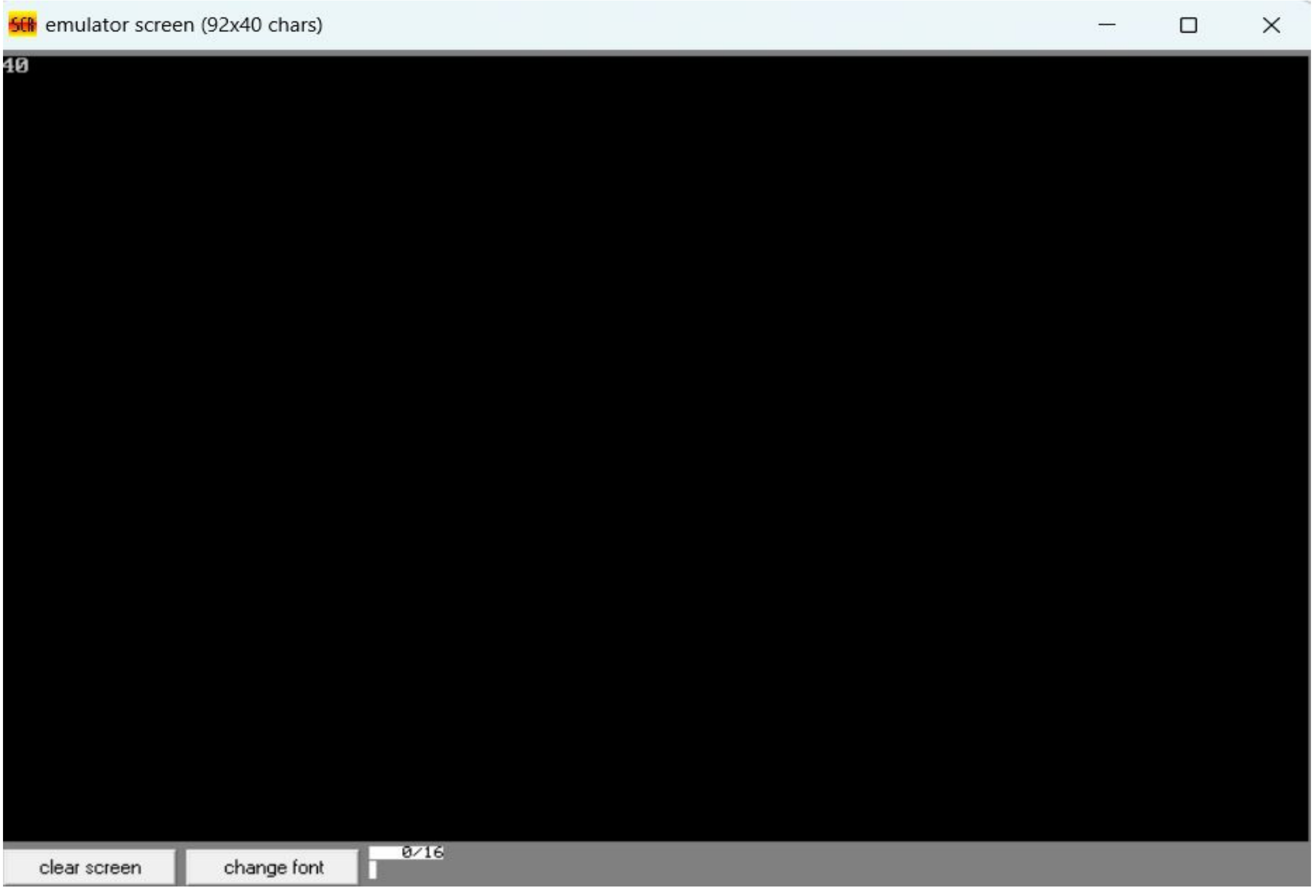
1. Write an assembly language program to perform multiplication of 8-bit data.

Code:

```
org 100h

mov al, 10h
mov bl, 4h
mul bl

mov bl, al
mov ah, al

and ah, 0F0h
shr ah, 4
add ah, 30h

cmp ah, 39h
jle print_first
add ah, 7

print_first:
mov dl, ah
mov ah, 02h
int 21h

mov ah, bl
and ah, 0Fh
add ah, 30h

cmp ah, 39h
jle print_sec
add ah, 7

print_sec:
mov dl, ah
mov ah, 02h
int 21h

mov ah, 4Ch
```

```
int 21h
```

output:



```
40
```

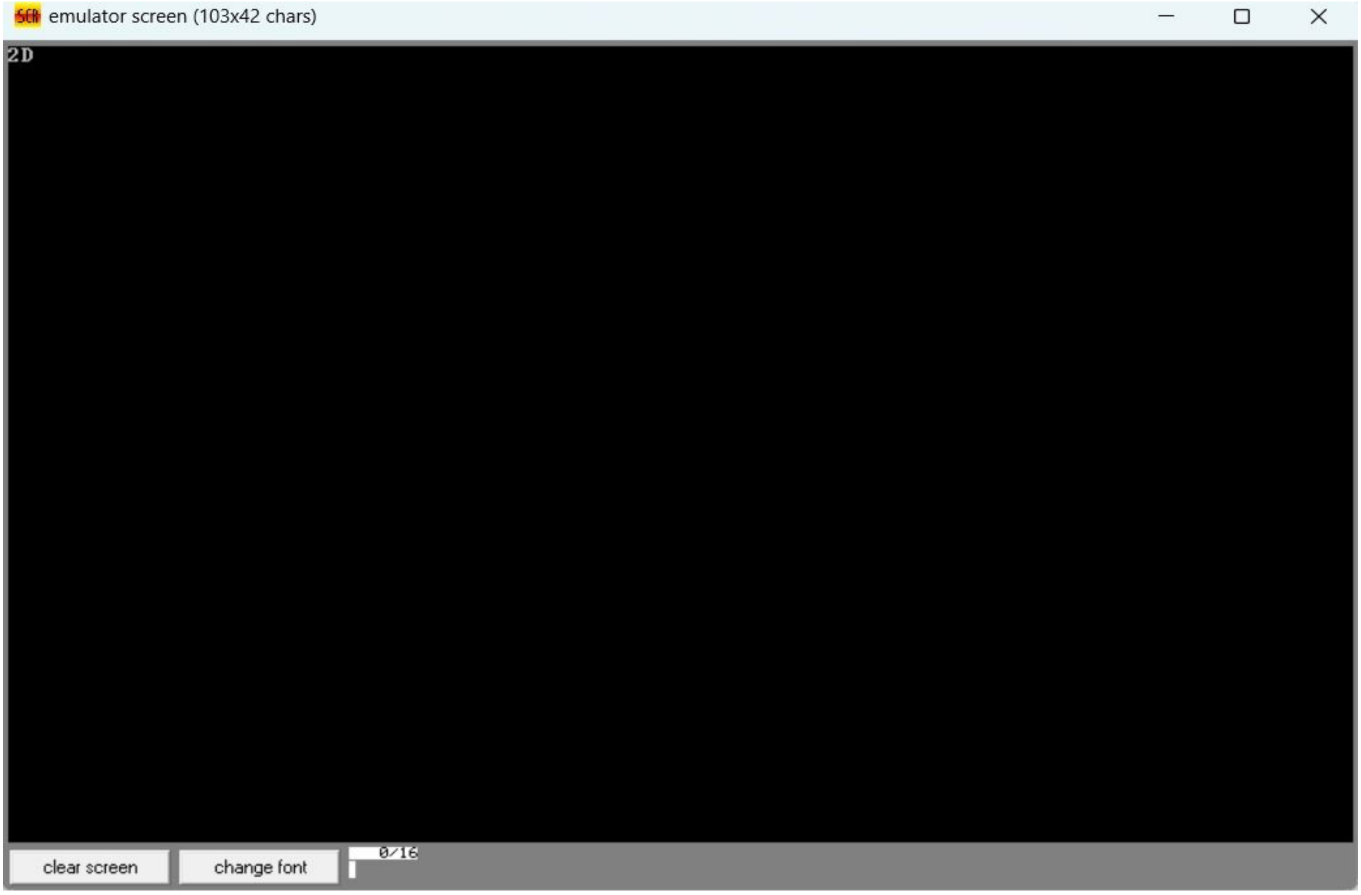
2. Write a program in assembly language to perform multiplication of 16-bit data.

Code:

```
org 100h
mov al,09h        ; AL = 9 (new input)
mov bl,05h        ; BL = 5 (new input)
mul bl          ; AL = AL * BL = 9 * 5 = 45 (decimal)

mov bl, al        ; BL = AL (store result)

mov ah, al        ; Move AL result to AH for extraction
and ah, 0F0h      ; Mask upper nibble
```

```
shr ah,4          ; Shift upper nibble
add ah, 30h       ; Convert to ASCII

cmp ah, 39h       ; Check if it's a digit
jle print_first_digit
add ah, 7         ; Adjust for letters A-F if necessary

print_first_digit:
mov dl,ah         ; DL = upper nibble
mov ah,02h        ; DOS interrupt to print character
int 21h           ; Print upper nibble

mov ah, bl        ; Restore full result in AH
and ah, 0Fh       ; Mask lower nibble
add ah, 30h       ; Convert to ASCII

cmp ah, 39h       ; Check if it's a digit
jle print_sec_digit
add ah,7          ; Adjust for letters A-F if necessary

print_sec_digit:
mov dl,ah         ; DL = lower nibble
mov ah, 02h       ; DOS interrupt to print character
int 21h           ; Print lower nibble

mov ah,4Ch        ; Terminate program
int 21h
```

output:



```
2D
```