UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA

Facultad de Ingeniería, Arquitectura y Diseño.

Ingeniería en Software y Tecnologías Emergentes

Organización de Computadoras

Taller 10

Alejandro Palomares Ceseña

1.-

HelloWorld.asm — 445devtqy — AI NEW ASSEMBLY RUN

```
section .data
char db 0
newline db 10

section .text
global _start

_start:

    mov al, 0x10        ; 00010000b
    shl al, 1           ; 00100000b = 0x20
    or  al, 0x01        ; 00100001b = 0x21
    rol al, 1           ; 01000010b = 0x42
    shr al, 1           ; 00100001b
    rol al, 1           ; 01000010b
    shr al, 1           ; 00100001b
    shl al, 1           ; 01000010b = 'B'
    ror al, 1           ; 00100001b
    rol al, 1           ; 01000010b
    sub al, 1           ; 0x41 = 'A'

    mov [char], al

    ; Output
    mov eax, 4
    mov ebx, 1
    mov ecx, char
    mov edx, 1
    int 0x80

    mov eax, 4
    mov ebx, 1
    mov ecx, newline
    mov edx, 1
    int 0x80

    mov eax, 1
    xor ebx, ebx
    int 0x80
```

STDIN: Input for the program (Optional)

Output:

```
A
```

2.-

HelloWorld.asm — 445devtqy — AI NEW ASSEMBLY RUN

```
section .data
char db 0
newline db 10

section .text
global _start

_start:

    mov al, 0x06        ; 00000110b
    shl al, 3           ; 00110000b = 0x30
    ror al, 1           ; 00011000b
    rol al, 1           ; 00110000b = '0'

    mov [char], al

    ; Output
    mov eax, 4
    mov ebx, 1
    mov ecx, char
    mov edx, 1
    int 0x80

    mov eax, 4
    mov ebx, 1
    mov ecx, newline
    mov edx, 1
    int 0x80

    mov eax, 1
    xor ebx, ebx
    int 0x80
```

STDIN: Input for the program (Optional)

Output:

```
0
```

3.-

HelloWorld.asm — 445devtqy — AI NEW ASSEMBLY RUN

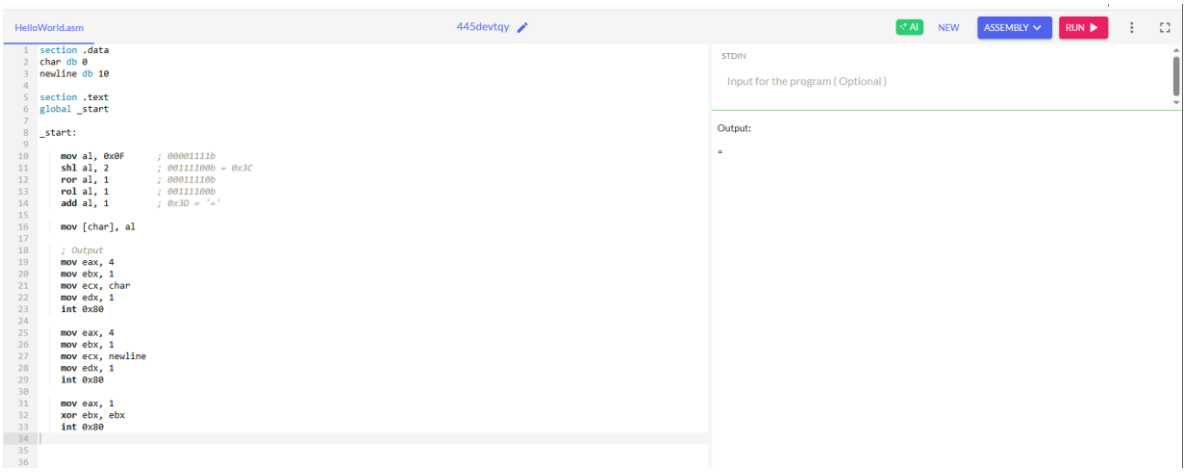
```
section .data
char db 0
newline db 10

section .text
global _start

_start:

    mov al, 0x19        ; 00011001b = 25
    shl al, 1           ; 00110010b = 0x32
    rol al, 1           ; 01100010b = 0x62 ('b')
    add al, 5           ; 0x67 = 'g'

    mov [char], al

    ; Output
    mov eax, 4
    mov ebx, 1
    mov ecx, char
    mov edx, 1
    int 0x80

    mov eax, 4
    mov ebx, 1
    mov ecx, newline
    mov edx, 1
    int 0x80

    mov eax, 1
    xor ebx, ebx
    int 0x80
```

STDIN: Input for the program (Optional)

Output:

```
i
```

4.-




```
section .data
char db 0
newline db 10

section .text
global _start

_start:

    mov al, 0x0F        ; 00001111b
    shl al, 2           ; 00111100b = 0x3C
    ror al, 1           ; 00011110b
    rol al, 1           ; 00111100b
    add al, 1           ; 0x3D = '='

    mov [char], al

    ; Output
    mov eax, 4
    mov ebx, 1
    mov ecx, char
    mov edx, 1
    int 0x80

    mov eax, 4
    mov ebx, 1
    mov ecx, newline
    mov edx, 1
    int 0x80

    mov eax, 1
    xor ebx, ebx
    int 0x80
```