

Universidad Nacional de Luján
41406 - Arquitectura de Computadoras
2º Parcial – 15/11/2017

1. Realice un programa que permita ingresar un caracter por teclado, y busque en la cadena CADENA si el valor aparece (El fin de CADENA se indica con el símbolo "\$"); de ser así que muestre por pantalla la cantidad de ocurrencias del mismo.

```
.model small
.stack
.data
    cartel1 db "Presione una tecla para buscar: $"
    cartel2 db "Apariciones :$"
    cadena db "qwertyuiopasdffgjl1234567890$"

.code
    mov ax, @data
    mov ds, ax
    xor si, si;Indice
    xor ax, ax
    xor bx, bx
    mov ah, 09h
    mov bx, offset cartel1
    int 21h
    mov ah, 01h
    int 21h
inicio:
    cmp cadena[si], '$'
    je fin
    cmp cadena[si], al
    jne otro
    inc bl
    otro:
    inc si
jmp inicio
fin:
    call nueva_linea
    add bl, 30h
    mov ah, 09h
    mov dx, offset cartel2
    int 21h
    mov dl, bl
    mov ah, 02h
    int 21h
    mov ah, 4ch
    int 21h

nueva_linea proc
    mov ah, 02h
    mov dl, 10
    int 21h
```

```

mov dl, 13
int 21h
ret
nueva_linea endp

```

2. Realice un programa que permita ingresar una cantidad ilimitada de números por teclado, cuyo caracter de escape es el ENTER, y muestre por pantalla el mayor de los números impares ingresados en binario.

```

.model small
.stack
.data
    cartel db "Ingrese el numero: $"
    cartel2 db "El numero en binario es: $"
.code
    mov ax, @data
    mov ds, ax
    xor ax, ax
    xor bx, bx
    mov dx, offset cartel; Muestro cartel para pedir numero
    mov ah, 09h
    int 21h
call ingresar
call nueva_linea
    mov dx, offset cartel2; Muestro cartel para mostrar binario
    mov ah, 09h
    int 21h
call mostrar
mov ah, 4ch
int 21h

```

ingresar proc ; Procedimiento ingresar dato

```

nuevo_numero:
    mov ah, 01h
    int 21h
    cmp al, 13
    je fin
    cmp al, 30h
    jb nuevo_numero
    cmp al, 39h
    ja nuevo_numero
    sub al, 30h
    mov ah, al
    and ah, 01h
    cmp ah, 00h
    je sigo
    cmp bh, 01h
    je impar
    inc bh
    mov bl, al
    impar:

```

```
    cmp al, bl
    jna sigo
    mov bl, al
sigo:
    jmp nuevo_numero
fin:
    ret
ingresar endp
```

mostrar proc ; Procedimiento mostrar valor en binario

```
    xor cx, cx
    mov cl, 8
    mov ah, 02h
muestra:
    mov dl, 30h
    rcl bl, 1
    adc dl, 00h
    int 21h
    loop muestra
    ret
mostrar endp
```

nueva_linea proc; Procedimiento para nueva linea

```
    mov ah, 02h
    mov dl, 10
    int 21h
    mov dl, 13
    int 21h
    ret
nueva_linea endp
```