DidadoY

Práctica Sesión 4

BUSCARELEMENTO.S

```
 .text
    .align 4
    .globl BuscarElemento
    .type BuscarElemento,@function
BuscarElemento:

          pushl %ebp
          movl %esp, %ebp
          pushl %ebx
          pushl %esi
          movl 24(%ebp), %ebx          #X.k
          movl 16(%ebp), %eax          #@mid
          movl (%eax), %eax    #mid
          movl 32(%ebp), %ecx          #ecx <- @v
          imull $12, %eax         #12**mid
          addl %eax, %ecx       #@v+12**mid = V[*mid] -> ecx
          cmpl 4(%ecx), %ebx
          jne else1

if1:
          movl 16(%ebp), %eax  # eax <- @mid
          movl (%eax), %eax    # eax <- *mid
          jmp endif1
else1:
          movl 16(%ebp), %ecx  # ecx <- @mid
          movl (%ecx), %ecx    # ecx <- *mid
          movl 12(%ebp), %edx  # edx <- @high
          movl (%edx), %edx    # edx <- *high
          cmpl %edx, %ecx
          jge else2
if2:
          movl 16(%ebp), %eax  # eax <- @mid
          movl %edx, (%eax)    # *mid=*high
          movl 8(%ebp), %esi  # esi <- @low
          movl (%esi), %esi       # esi <- *low
          incl %esi
          movl 8(%ebp), %edx   #@low
          movl %esi, (%edx)
          jmp endif2
else2:
          movl 8(%ebp), %ecx  #@low
          movl (%ecx), %ecx   #*low
```

```
        movl 16(%ebp), %eax  # eax <- @mid
        movl %ecx, (%eax)
        decl %edx
        movl 12(%ebp), %ecx
        movl %edx, (%ecx)
endif2:
        movl $-1, %eax

endif1:
        popl %esi
        popl %ebx
        movl %ebp, %esp
        popl %ebp
        ret
```

BUSCAR.S

```
 .text
   .align 4
   .globl Buscar
   .type Buscar,@function
Buscar:
    # Aqui viene vuestro codigo
        pushl %ebp
        movl %esp, %ebp
        subl $16, %esp          # variables locales (4 ints)
        movl $-1, -4(%ebp)      # trobat = -1
        movl $0, -16(%ebp)      # low = 0
        movl $0, -8(%ebp)       # mid = 0
        movl 24(%ebp), %eax     # N
        subl $1, %eax           # N-1
        movl %eax, -12(%ebp)    # high = N-1
while:  movl -16(%ebp), %edx    # low
        movl -12(%ebp), %eax    # high
        cmpl %eax, %edx
        jg fiwhile          # fiwhile si low > high
        pushl 8(%ebp)           # pushl @v
        pushl 20(%ebp)          # X.m
        pushl 16(%ebp)          # X.k
        pushl 12(%ebp)          # X.c
        leal -8(%ebp), %eax
        pushl %eax      # &mid
        leal -12(%ebp), %eax
        pushl %eax      # &high
        leal -16(%ebp), %eax
        pushl %eax      # &low
        call BuscarElemento
        addl $28, %esp          # eliminar parametros: 7 pushl's, 28 (4 + 12 + 4 + 4 + 4)
        movl %eax, -4(%ebp)     # trobat = BuscarElemento(...)
if:     cmpl $0, %eax
        jl while            # si < while; si >=, fiwhile
```

```
fiwhile: movl -4(%ebp), %eax
         movl %ebp,%esp
         popl %ebp
         ret
```