**Ejercicio de Repaso – Módulo Assembler**

1. Implemente una subrutina en Assembler que reciba un parámetro A por referencia vía pila

y un parámetro B por valor a través de registro y reemplace el valor de A, sólo si el valor de B **es**

**menor** al de A.

1. Implemente un programa en Assembler que, utilizando la subrutina del inciso a), reemplace los valores

de la TablaA por los de la TablaB cuando corresponda (posición a posición). Por ejemplo, si tenemos las siguientes tablas:

* TablaA 🡪 1, 2, 5, 4, 7,…
* TablaB 🡪 2, 3, 4, 5, 7,…

los valores finales serán: TablaA 🡪 1, 2, 4, 4, 7,…

Además, el programa debe sumar **eficientemente** los valores de TablaB mayores a 5y almacenar este

resultado en memoria principal.

Código:

ORG 1000H ;Datos

TablaA DB 1, 2, 5, 4, 7

TablaB DB 2, 3, 4, 5, 7

Resul DB ?

ORG 3000H ; Subrutina

Subr: PUSH BX

PUSH AX

MOV BX,SP ;traemos el puntero al tope de la pila

ADD BX,6 ;sumo & para saltar el Dir. de retorno y push de AX y BX

MOV BX,[BX] ;Nos traemos la dir. del parámetro

MOV AH,[BX] ;Recibir un par. por referencia via pila (16b) valor A

;Recibir u par. por valor vía registro (8bits) en AL, valor B

CMP AL,AH ; resta de AL-AH B-A

JNS FinR

reemplazar: MOV [BX],AL ; reemplazo

FinR: POP AX

POP BX

RET

ORG 2000H ;Programa

MOV CX, OFFSET TablaA ; dir. de TablaA

MOV BX, OFFSET TablaB ; dir. de TablaB

MOV DL, OFFSET TablaB - OFFSET TablaA ; cantidad de elementos de las tablas

MOV DH, 0 ;TOTALIZADOR

bucle:PUSH CX; Mandar via pila la dir. de la TablaA

MOV AL,[BX]; Mandar el dato de la subrutina en AL (via registro)

CALL Subr

POP CX

CMP AL, 6

JS Sigue

ADD DH,AL; suma del valor de TablaB

Sigue: INC CX ; para avanzar en la TablaA

INC BX ; para avanzar en la TablaB

DEC DL

JNZ bucle

MOV Resul, DH; guardamos la suma

Fin: HLT

END