**LABORATORIO NO. 02**

“Ensamblador, DEBUG y Sistemas Numéricos”

**Ejercicio 1: Utilización del Ensamblador y el Enlazador**

Utilizando los archivos “Ejemplo1.asm” y “Ejemplo2.asm” genere el código objeto y el programa ejecutable utilizando el Ensamblador “TASM” y el Enlazador “TLINK”.

Conteste las siguientes preguntas:

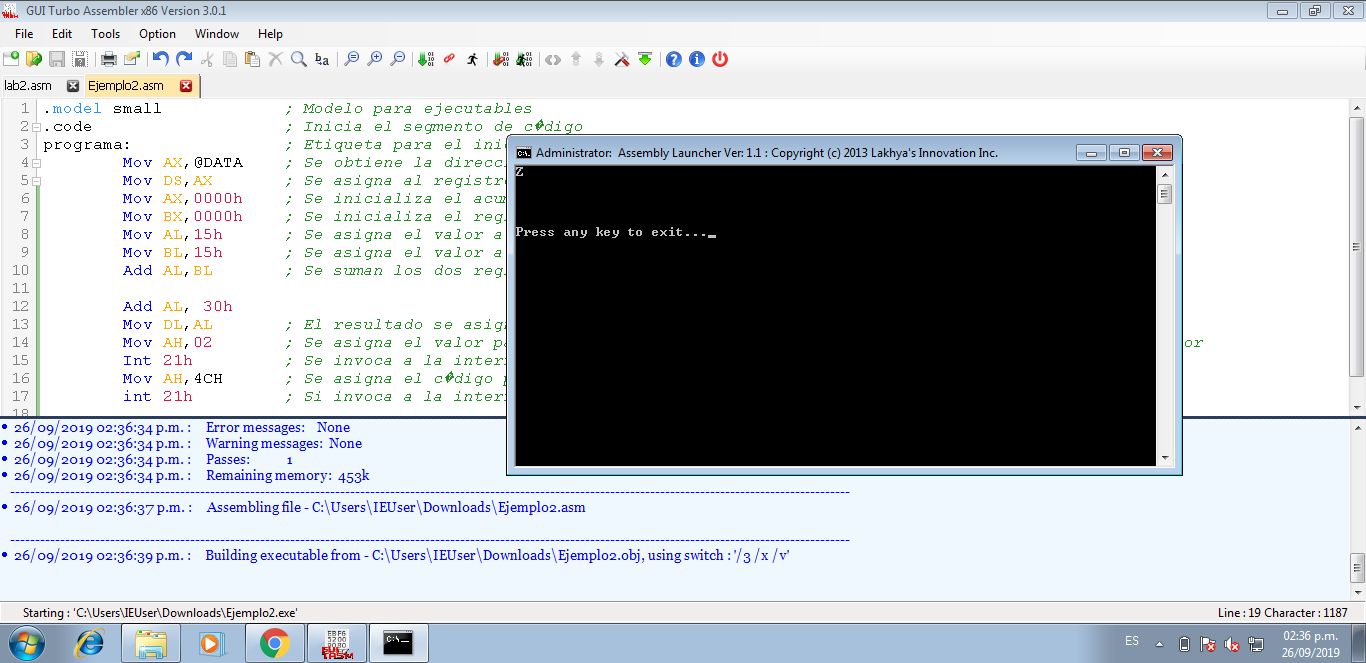
1. Cuando se genera el ejecutable del archivo “Ejemplo2.asm”, ¿cuál es la advertencia que se muestra en pantalla? ¿Por qué muestra esa advertencia?

R// Pone la advertencia porque no se declaro el segmento de stack en el código

1. Modificado el programa para que no muestre la advertencia, ¿cuál es el resultado del programa, es decir, por qué se imprime ese carácter y no un 30?

R// Porque es el numero en hexadecimal de la letra z

1. Modifique el código del archivo “Ejemplo2.asm” y utilizando la tabla de códigos ASCII, imprima en pantalla una letra “Z”.



**Ejercicio 2: Utilización del Modo “DEBUG”**

Comandos del Modo “DEBUG”:

* N Nombrar un programa.
* L Se encarga de cargar el programa.
* U        "Desensamblar" código máquina y pasarlo a código simbólico.
* A        Ensamblar instrucciones simbólicas y pasarlas a código máquina.
* D        Mostrar el contenido de un área de memoria.
* E        Introducir datos en memoria, iniciando en una localidad específica.
* G       Correr el programa ejecutable que se encuentra en memoria.
* P        Proceder o ejecutar un conjunto de instrucciones relacionadas.
* Q       Salir de la sesión con DEBUG.
* R       Mostrar el contenido de uno o más registros.
* T        Rastrear la ejecdebución de una instrucción.
* W       Escribir o grabar un programa en disco.

Utilizando el Modo “DEBUG” de DOS cargue el programa “Ejemplo2.exe” y responda las siguientes preguntas: t

1. ¿En qué dirección de memoria inicia el código del programa?

R//0B6C:0000 B86D0B MOV AX, 0B6D

1. ¿En qué dirección de memoria termina el código del programa?

R//0B6C:001F 00FB ADD BL, BH

1. Aparecen los comentarios en pantalla ¿Sí? ¿No? ¿Por qué?

R//No porque no son procedimientos ni etiquetas es algo que para el compilador es indiferente

1. Para cada una de las instrucciones del programa, escriba la dirección de memoria que tiene asignada:

|  |  |
| --- | --- |
| Dirección de memoria | Instrucción |
| 0B6C:0000 B86D0B | Mov AX,@DATA |
| 0B6C:0003 8ED8 | Mov DS,AX |
| 0B6C:0005 B80000 | Mov AX,0000h |
| 0B6C:0008 BB0000 | Mov BX,0000h |
| 0B6C:000B B015 | Mov AL,15h |
| 0B6C:000D B315 | Mov BL,15h |
| 0B6C:000F 02C3 | Add AL,BL |
| 0B6C:0013 8AD0 | Mov DL,AL |
| 0B6C:0013 b492 | Mov AH,02 |
| 0b6c:001B CD21 | Int 21h |
| DD  A  D 0B6C:0019 B44C | Mov AH,4CH |
| 0B6C:001B CD21 | int 21h |

1. ¿Cuál es la dirección del segmento de código?

CS= 0B6C

1. Antes de iniciar la ejecución por pasos del programa, ¿cuáles son los valores de los registros de propósito general?

AX=0000

BX=0000

CX=01E2

DX=0000

1. El valor del IP, ¿coincide con la dirección de inicio del programa?

Ambas coinciden porque es 0B6C

1. Utilice el comando para el rastreo instrucción por instrucción y, por cada línea del código, escriba debuel contenido de los registros internos del CPU.

0B6C:0000 B86D0B MOV AX,0B6D

AX=0000  
BX=0000  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B5C  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0000

0B6C:0003 8ED8 MOV DS,AX

AX=0B6D  
BX=0000  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B5C  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0003

0B6C:0005 B80000 MOV AX,0000

AX=0B6D  
BX=0000  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0005

0B6C:0008 BB0000 MOV BX,0000

AX=0000  
BX=0000  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0008

0B6C:000B B015 MOV AL,15

AX=0000  
BX=0000  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6d  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=000B

0B6C:000D B315 MOV BL,15

AX=0015  
BX=0000  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=000D

0B6C:000F 02C3 ADD AL,BL

AX=0015  
BX=0015  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=000F

0B6C:0000 B86D0B ADD AL,30

AX=002A  
BX=0015  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0011

0B6C:0013 8AD0 MOV DL,AL

AX=005A  
BX=0015  
CX=01E2  
DX=0000  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0013

0B6C:0015 B402 MOV AH,02

AX=005A  
BX=0015  
CX=01E2  
DX=005A  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0015

0B6C:0017 CD21 INT 21

AX=025A  
BX=0015  
CX=01E2  
DX=005A  
SP=0000  
BP=0000  
SI=0000  
DI=0000  
DS=0B6D  
ES=0B5C  
SS=0B6C  
CS=0B6C  
IP=0017