|  |  |
| --- | --- |
| Francisco Josué Solís Ruano | 1050014 |

**LABORATORIO NO. 01**

“Ensamblador, DEBUG y Sistemas Numéricos”

**Objetivo**

* Mostrar la estructura básica de un programa en Lenguaje ensamblador, así como la forma de ensamblarlo y generar el código objeto, además de su enlace como programa ejecutable.
* Demostrar el funcionamiento de la herramienta para realizar la depuración de un programa escrito en Lenguaje Ensamblador.
* Familiarizar los diversos sistemas numéricos, sus conversiones, operaciones y lógica apropiada para representar información en un programa en Lenguaje Ensamblador.

**Ejercicio 1: Declaración de variables e intercambio de valores**

Declarar dos variables, que almacenen dos valores,

* Variable 1: nombre del alumno
* Variable 2: carnet del alumno

Imprimir ambos valores, luego intercambiarlos, es decir, variable 1 = variable 2 y variable 2= variable 1, y de nuevo imprimir los valores.

**Ejercicio 2: 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?

**Warning: No Stack**

¿Por qué muestra esa advertencia?

**Porque no se definió la directiva de STACK.**

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?

**Porque se está sumando 15 + 15 pero en hexadecimal, cuyo resultado es igual a 2A, que en ASCII es equivalente a 42 que corresponde al símbolo del asterisco [\*].**

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

**Ejercicio 3: 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 ejecución 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:

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

**0B6D:0000 B86F0B**

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

**0B6D:0019 CD21**

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

**No aparecen, en el proceso de compilación se eliminan los comentarios.**

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

|  |  |
| --- | --- |
| **Dirección de memoria** | **Instrucción** |
| 0B6D:0000 B86F0B | Mov AX,@DATA |
| 0B6D:0003 8ED8 | Mov DS,AX |
| 0B6D:0005 B80000 | Mov AX,0000h |
| 0B6D:0008 BB0000 | Mov BX,0000h |
| 0B6D:000B B015 | Mov AL,15h |
| 0B6D:000D B315 | Mov BL,15h |
| 0B6D:000F 02C3 | Add AL,BL |
| 0B6D:0011 8AD0 | Mov DL,AL |
| 0B6D:0013 B402 | Mov AH,02 |
| 0B6D:0015 CD21 | Int 21h |
| 0B6D:0017 B44C | Mov AH,4CH |
| 0B6D:0019 CD21 | int 21h |

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

**0B6D:0000 B86F0B**

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0000 | SP | FFEE | DS | 0B05 |
| BX | 0000 | BP | 0000 | ES | 0B05 |
| CX | 0000 | SI | 0000 | SS | 0B05 |
| DX | 0000 | DI | 0000 | CS | 0B05 |
| IP | 0100 |

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

**Si, pero solo al comenzar a ejecutar el programa.**

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

Instrucción 1:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0B6F | SP | 0400 | DS | 0B5D |
| BX | 0000 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 0003 |

Instrucción 2:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0B6F | SP | 0400 | DS | 0B6F |
| BX | 0000 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 0005 |

Instrucción 3:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0000 | SP | 0400 | DS | 0B6F |
| BX | 0000 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 0008 |

Instrucción 4:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0000 | SP | 0400 | DS | 0B6F |
| BX | 0000 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 000B |

Instrucción 5:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0015 | SP | 0400 | DS | 0B6F |
| BX | 0000 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 000D |

Instrucción 6:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 0015 | SP | 0400 | DS | 0B6F |
| BX | 0015 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 000F |

Instrucción 7:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 002A | SP | 0400 | DS | 0B6F |
| BX | 0015 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 0000 | DI | 0000 | CS | 0B6D |
| IP | 0011 |

Instrucción 8:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 002A | SP | 0400 | DS | 0B6F |
| BX | 0015 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 002A | DI | 0000 | CS | 0B6D |
| IP | 0013 |

Instrucción 10:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 022A | SP | 0400 | DS | 0B6F |
| BX | 0015 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 002A | DI | 0000 | CS | 0B6D |
| IP | 0015 |

Instrucción 11:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 022A | SP | 0400 | DS | 0B6F |
| BX | 0015 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 002A | DI | 0000 | CS | 0B6D |
| IP | 0017 |

Instrucción 12:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Registro** | **Valor** | **Registro** | **Valor** | **Registro** | **Valor** |
| AX | 4C2A | SP | 0400 | DS | 0B6F |
| BX | 0015 | BP | 0000 | ES | 0B5D |
| CX | 05E2 | SI | 0000 | SS | 0B6F |
| DX | 002A | DI | 0000 | CS | 0B6D |
| IP | 0019 |