

Tampico, Tamaulipas a **10** de **febrero** de **2023**

**Portafolio de Evidencias**

**“unidad 1”**

**programación de microprocesadores**

Profesor: **Dr. García Ruiz Alejandro Humberto**

**8**vo. Semestre – Grupo “**I**”

2023-1

**Autor de Entregas Individuales:**

Ruiz Garcia Emmanuel Alejandro – 2193330288

**Autores de Entregas en Equipo:**

Luna Sánchez Juan Pablo - 2163216015

**ÍNDICE**

[1. Actividades de Clase 3](#_Toc126857762)

[1.1 Entregas en Equipo 4](#_Toc126857763)

[1.2 Entregas Individuales 4](#_Toc126857764)

[2. Tareas e Investigaciones 5](#_Toc126857765)

[2.1 Entregas en Equipo 6](#_Toc126857766)

[2.2 Entregas Individuales 17](#_Toc126857767)

[3. Programas 18](#_Toc126857768)

[3.1 Entregas en Equipo 19](#_Toc126857769)

[3.2 Entregas Individuales 19](#_Toc126857770)

[4. Prácticas 20](#_Toc126857771)

[4.1 Entregas en Equipo 21](#_Toc126857772)

[4.1.1 Desarrollo de las Prácticas 21](#_Toc126857773)

[4.2 Entregas Individuales 22](#_Toc126857774)

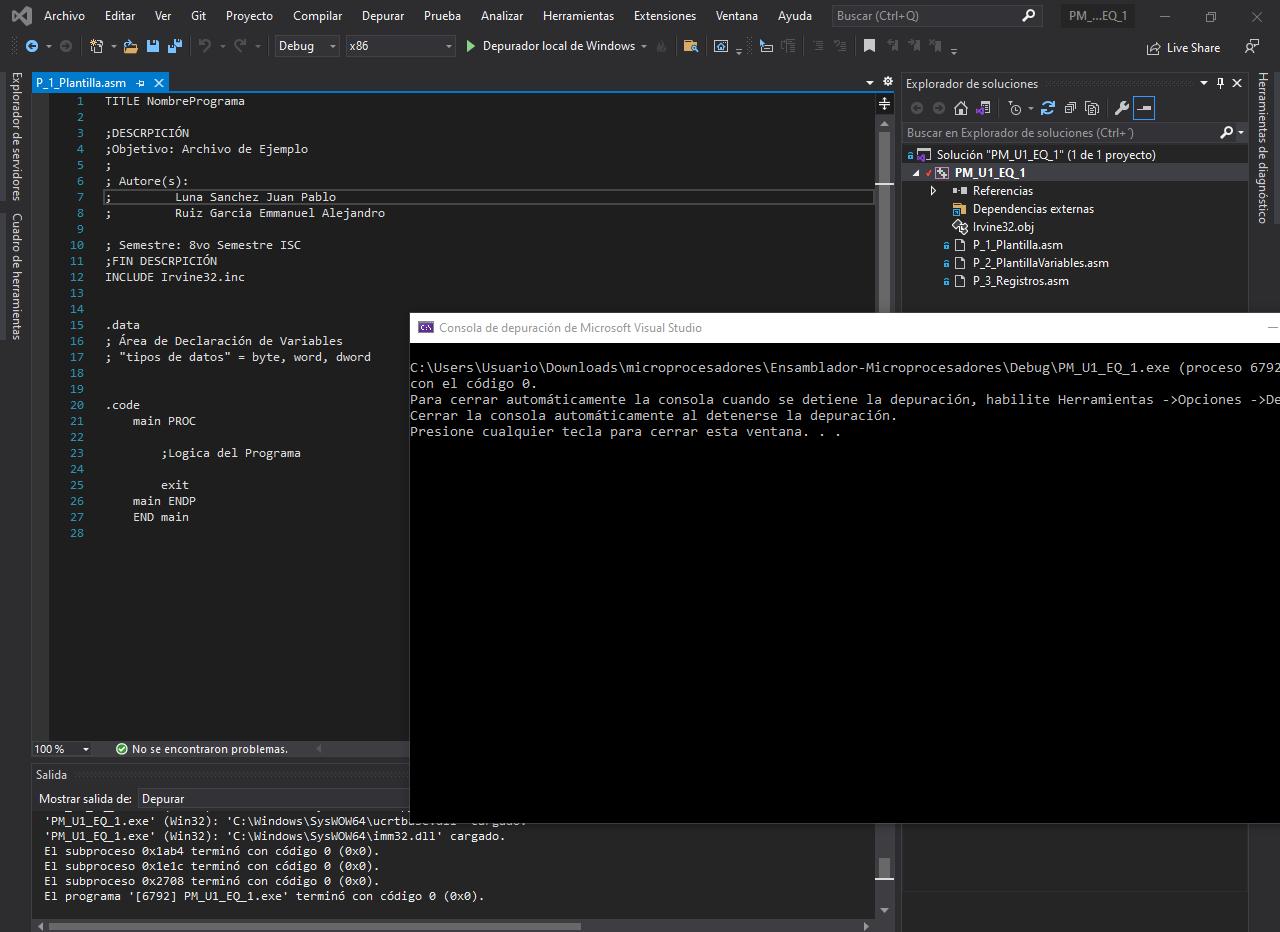
[5. Proyecto 23](#_Toc126857775)

# 1. Actividades de Clase

**Actividades de Clase**

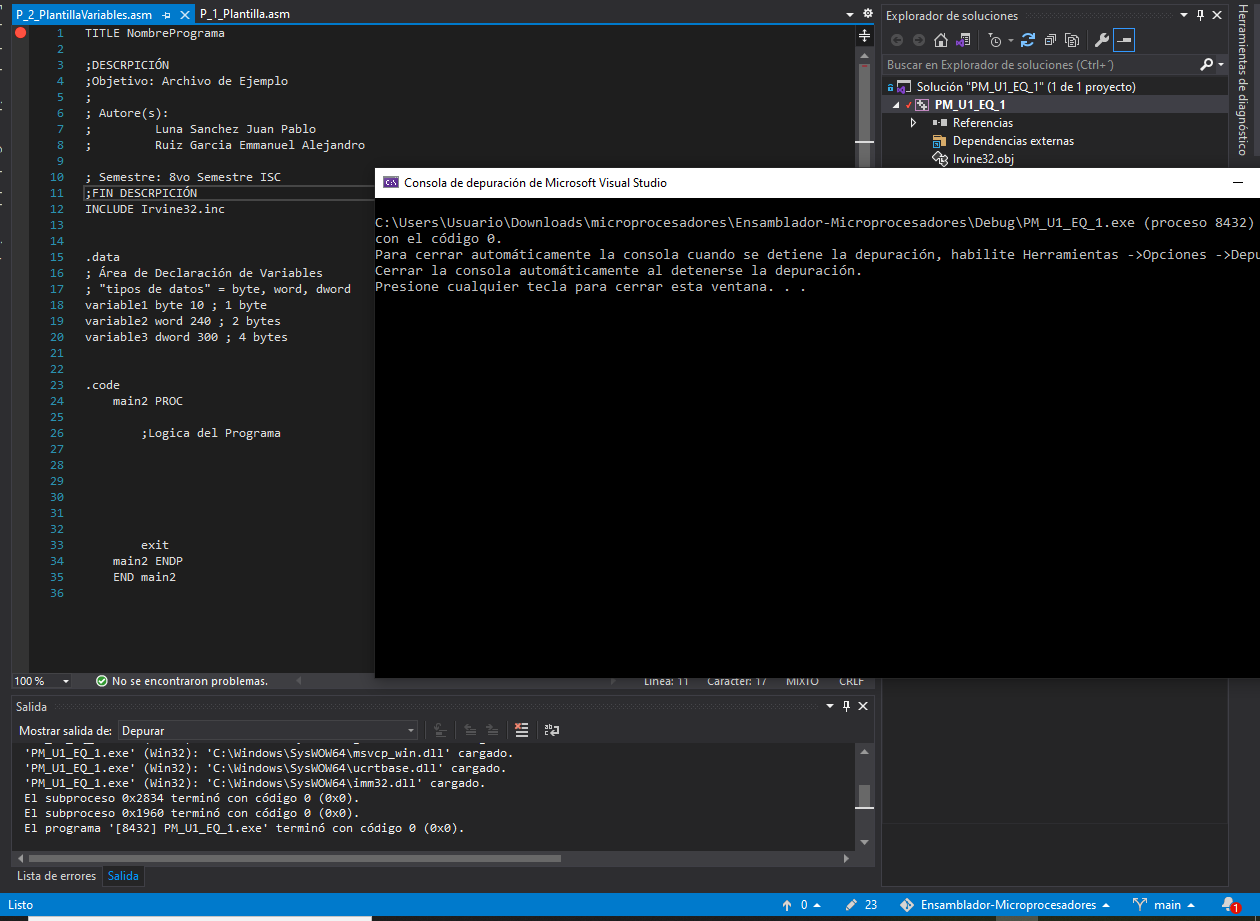
## 1.1 Entregas en Equipo

EJERCICIO 1.



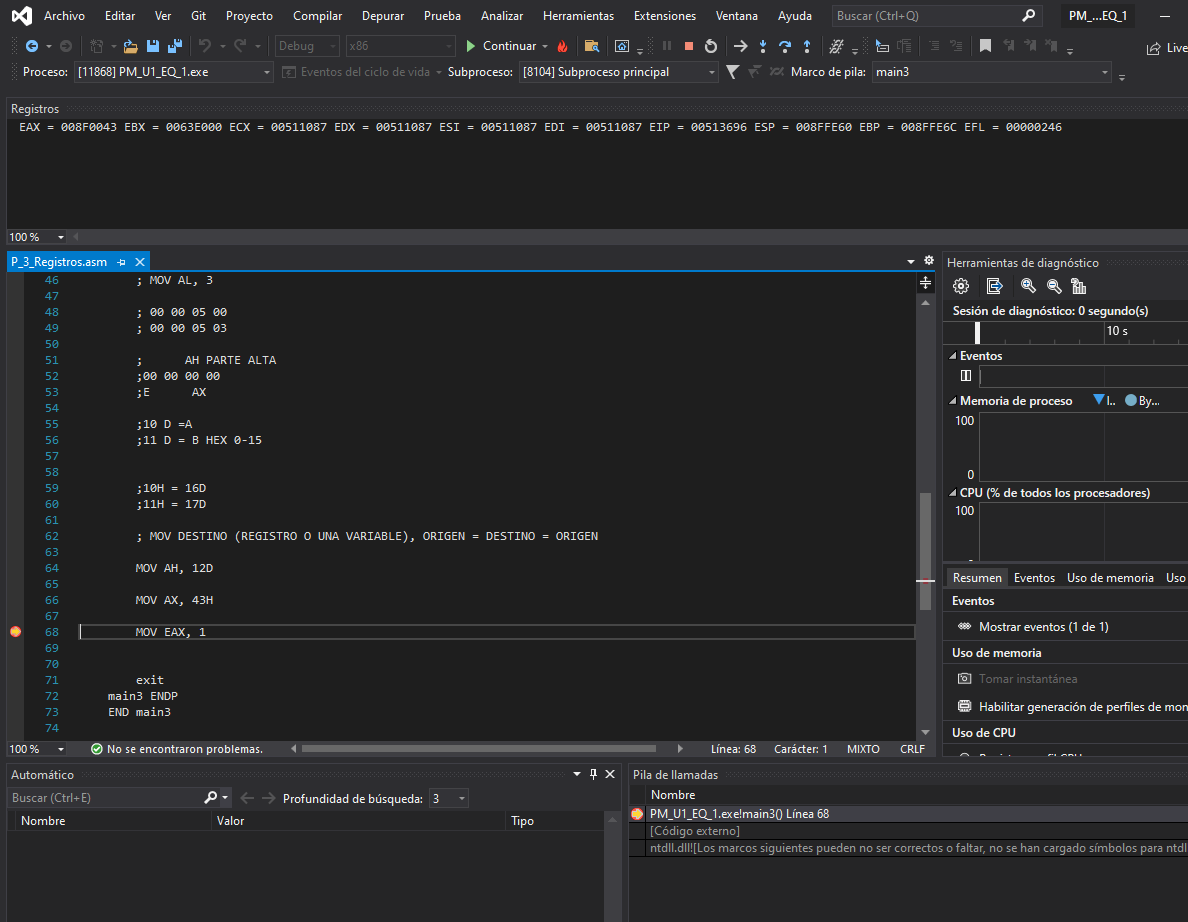
En esta actividad fue crear nuestro primer archivo o elemento en este caso fue una plantilla para poder ver que todo estaba funcionando correctamente

EJERCICIO 2.



En la siguiente actividad fue el ver como se crean variables y los tipos de datos que se utilizan.

EJERCICIO 3.



En la siguiente actividad trabajamos con la instrucción mov para dar valores el cual aprendimos que el mov va hacia un destino, además en el apartado de registros se pueden observar los valores que tenemos en Eax.

## 1.2 Entregas Individuales

No aplica en esta unidad

# 2. Tareas e Investigaciones

**Tareas e Investigaciones**

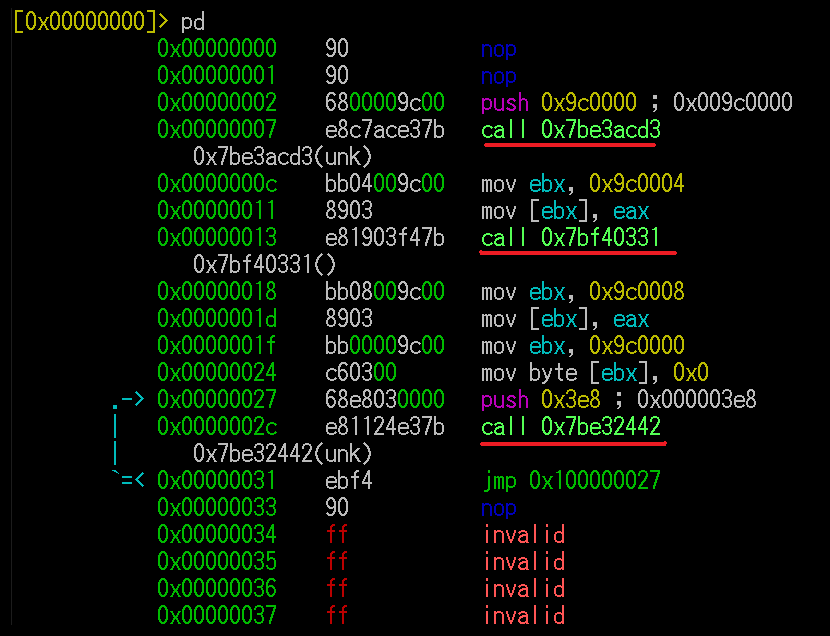
## 2.1 Entregas en Equipo

**1. ¿Porque se sigue utilizando lenguaje ensamblador?**

El lenguaje ensamblador sigue siendo importante, ya que ofrece al programador el control total de la máquina y como resultado genera un código compacto, rápido y eficiente.

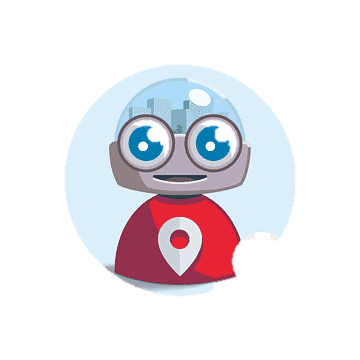
No se usa para desarrollar proyectos completos (que hoy en día requieren miles y millones de líneas de código). Se usa para...

* Tareas específicas que deben ejecutarse con máxima eficiencia (alta velocidad, bajo uso de recursos)
* Interactuar con hardware y dispositivos periféricos
* Programar microcontroladores
* En docencia, para comprender al más bajo nivel el funcionamiento del procesador
* Es necesario comprenderlo para incursionar en hacking y cracking



El lenguaje ensamblador es ideal para escribir programas embebidos, debido a que utilizan muy poca memoria. El uso más común sin duda es la programación de microcontroladores, para diseñar sistemas embebidos, el cual es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria.

**2. 5 trabajos que pidan ensamblador**



**1.- SOLICITAMOS ENSAMBLADORES PARA EMPRESA LÍDER EN AUTOMATIZACIÓN INDUSTRIAL**  
  
**REQUERIMOS:**  
-Experiencia a partir de 1 año en el puesto  
-Habilidad manual  
  
**ACTIVIDADES PRINCIPALES:**  
-Corte y desforre de cables  
-Armado de arneses eléctricos  
-Ensamble de conectores y pines eléctricos  
-Sujeción de cables con cinchos para procesos de manufactura  
  
**BRINDAMOS:**  
-Sueldo de $9,500 mensuales brutos  
-Prestaciones de ley  
  
**HORARIO DE TRABAJO:** Lunes a viernes de 08:00 am a 06:00 pm  
**ZONA DE TRABAJO:** Tlalnepantla

INTERESADOS POSTULARSE POR ESTE MEDIO O COMUNICARSE AL\*PARA AGENDAR UNA ENTREVISTA

* Nivel de educación deseada: Media Superior
* Nivel de experiencia deseada: Nivel Medio
* Función departamental: Manufactura / Operaciones
* Tipo de empleo: Tiempo completo
* Estado: Estado de México

2.- **Técnico Ensamblador de PC**

Cyberpuerta S.A. de C.V.

5 evaluaciones

Tlaquepaque, Jal.

$12,000 a $13,000 por mes - Tiempo completo, Por tiempo indeterminado

**Detalles del empleo**

Salario

$12,000 a $13,000 por mes

Tipo de empleo

Tiempo completo

Por tiempo indeterminado

**Cualificaciones**

* Ensamble de equipos de cómputo: 1 año (Obligatorio)

Descripción completa del empleo

**Técnico ensamblador PC**

**¡Continuamos creciendo!**

Tras la inauguración inauguramos nuestro laboratorio de ensamble de PC’s y tras el éxito obtenido con esta nueva área y servicio, nos encontramos en la búsqueda de dos nuevos integrantes para crecer este equipo.

Buscamos técnicos altamente capacitado para armar las computadoras de diferentes gamas que nuestros clientes requieren y lograr así crecer nuestra presencia en este sector del mercado de la electrónica en México. Las principales responsabilidades del puesto son el ensamblaje de PC´s basado en especificaciones técnicas, con alta calidad y atención a detalles.

**Tu perfil**

* Radicas en Guadalajara
* Tienes experiencia probada en ensamble de PC´s.
* Te gusta el armado de PC´s y la tecnología.
* Eres ágil para el ensamble de PC y te apasiona hacer un trabajo de impecable calidad.
* Eres una persona meticulosa, ordenada y con atención a los detalles.
* Cuentas con conocimientos técnicos de ensamble de PC´s.
* Te gustan los retos y el armado de PC gaming y de alta gama.
* Te apasiona la tecnología y el comercio electrónico

**Beneficios de ser parte del equipo:**

* Horario laboral de Lunes a Viernes de 09:00 am a 7:00 pm
* 1 hora de comida + 1 hora de pausas flexibles al día
* 5 días de vacaciones a partir del sexto mes + 12 a tu primer aniversario
* Seguro de Vida
* Fondo de ahorro
* Póliza de accidentes
* Aguinaldo de 22.5 días por año
* Reparto de utilidades anual
* Contratación directa por parte de la empresa
* Descuento de empleado en todos los artículos de Cyberpuerta
* Permisos adicionales con goce de sueldo
* Contrato por tiempo indeterminado desde el primer día

**Sobre nosotros**

* Somos orgullosamente la empresa #1 de E-Commerce en México en productos tecnológicos, con presencia en toda la república mexicana y un portafolio de más de 45,000 productos.
* Somos una empresa nativa digital consolidada, conformada por un equipo experimentado, con más de 14 años en el mercado
* Contamos con un ambiente de trabajo moderno, competitivo y de excelencia que favorece tu desarrollo profesional
* Nuestra cultura organizacional está avalada por el prestigioso instituto Great Place To Work lo cual nos reconoce como de una de las mejores empresas para trabajar en México

**Consideraciones:**

* La zona de trabajo es el álamo industrial. (Álamo Business Park)
* Por la índole del puesto, la modalidad de trabajo es 100% presencial; sin embargo, contamos con medidas de seguridad e higiene para cuidar tu salud.

Tipo de puesto: Tiempo completo, Por tiempo indeterminado

Salario: $12,000.00 - $13,000.00 al mes

Horario:

* Horario flexible
* Lunes a viernes
* Turno de 8 horas

3.- **Ensamblador De Tarjetas Electrónicas**

Data Móvil Systems

Conocimientos básicos de soldadura, ensamble de tarjetas electrónicas.Reparación de tarjetas electrónicas ( no inspensable),uso de multimetro y cautín.Tipo de puesto: Tiempo completoSalario: A partir de $1,800.00 a la semanaHorario: Turno de 8 horasPrestaciones: Aumentos salariales Días por enfermedad Estacionamiento de la empresa Opción a contrato indefinido Seguro de gastos médicos Seguro de gastos médicos mayores Uniformes gratuitosEducación: Secundaria terminada (Deseable)Experiência: Electrónica o producción: 1 año (Deseable).

4.- **Técnico Ensamblador**

Descripción de la vacante

**Vacante para la empresa HUMANNEED en Tijuana, Baja California**: ¡IMPORTANTE EMPRESA DE SOPORTE Y MANTENIMIENTO A CASINOS, TE INVITA A FORMAR PARTE DE SU EQUIPO!!SOLO NECESITAS: Carrera técnica o Bachillerato concluido Experiencia de un año en labores de mantenimiento Conocimientos en electricidad, hojalatería, soldadura, pintura y plomería, colocación de tableros electrónicos. Conocimientos en software (programación) Conocimientos de manejo en herramientas (taladro, sierra, compresor, pistolas de aire, multímetro, cautín) OFRECEMOS: Sueldo base de $13500 brutos al mes Prestaciones superiores de ley Horario de Lunes a viernes. De 7 a 5 pm

**Nivel de educación deseada**: Media Superior

**Nivel de experiencia deseada**: Nivel Medio

**Función departamental**: Mantenimiento y reparaciones

**Industria**: Proovedores de Entretenimiento

**Habilidades**: proactivo buena disposición responsable

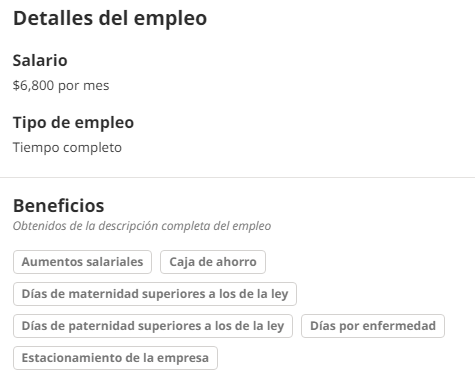
**5.- Ayudante General (Ensamblador)**

Schneider Electric

5,034 evaluaciones

Ciudad de México

$6,800 por mes - Tiempo completo



**Descripción completa del empleo**

Empresa Francesa líder en control y automatización de energía eléctrica a nivel mundial solicita por crecimiento:

Ayudantes Generales (Ensambladores). SALARIO $6800 MENSUALES

Requisitos:

\*Secundaria terminada  
\*Conocimiento básico en herramientas neumáticas  
\*Disponibilidad de rolar turnos.  
\*SIN/CON Experiencia

Tipo de puesto: Tiempo completo

Sueldo: $6,800.00 al mes

Horario:

* Rotativo

Prestaciones:

* Aumentos salariales
* Caja de ahorro
* Días de maternidad superiores a los de la ley
* Días de paternidad superiores a los de la ley
* Días por enfermedad
* Estacionamiento de la empresa
* Programa de referidos
* Servicio de comedor
* Uniformes gratuitos
* Vales de despensa

Tipos de compensaciones:

* Bono de asistencia
* Bono de productividad
* Bono de puntualidad

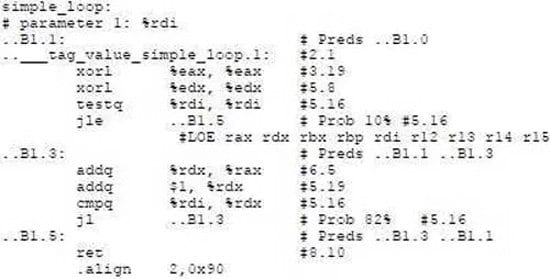
**3. Software que está escrito en lenguaje máquina y su funcionalidad**

El ensamblador es un programa, escrito en lenguaje de máquina, suministrado por el manufacturero, el cual instruye a la computadora sobre cómo convertir programas escritos en lenguaje simbólico a lenguaje de máquina.  Esta traducción es por lo general de uno a uno.

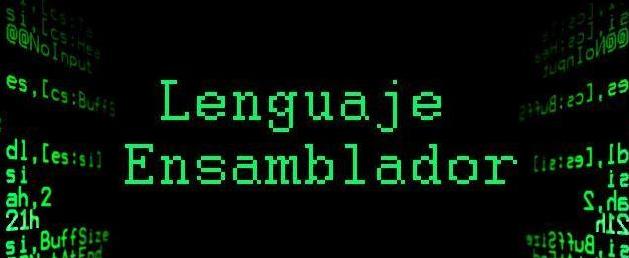
El lenguaje sobre el cual se profundiza en este artículo. El lenguaje ensamblador, o Assembly Language en inglés fue el primer intento en la búsqueda de un sustituto más sencillo de comunicarse de forma directa con el hardware de un dispositivo.

Cabe destacar en este punto que el número IEV de los programas del tipo ensamblador dado por la International Electrotechnical Commission es el 171-05-17, y está inscripto en la categoría “Digital technology – Fundamental concepts / Software”.

Este tipo de lenguajes actúan como conversores, ya que los dispositivos utilizan el lenguaje binario, y el programador un lenguaje de programación más flexible y sencillo de usar. Este es llamado lenguaje de alto nivel.

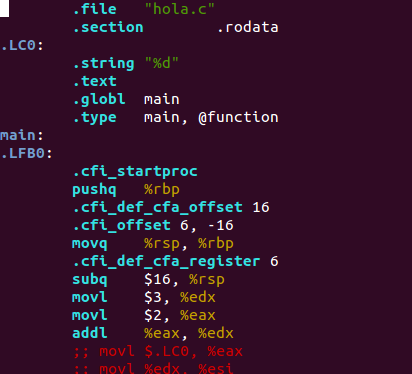


El término ensamblador (del inglés assembler) se refiere a un tipo de programa informático que se encarga de traducir un fichero fuente escrito en un lenguaje ensamblador, a un fichero objeto que contiene código máquina, ejecutable directamente por el microprocesador.



**4. Características, ventajas y desventajas del ensamblador**

Consiste en un conjunto de mnemónicos que representan instrucciones básicas para los computadores, microprocesadores, microcontroladores y otros circuitos integrados programables. Implementa una representación simbólica de los códigos de máquina binarios y otras constantes necesarias para programar una arquitectura de procesador y constituye la representación más directa del código máquina específico para cada arquitectura legible por un programador.



El código escrito en lenguaje ensamblador posee una cierta dificultad de ser entendido ya que su estructura se acerca al lenguaje máquina, es decir, es un lenguaje de bajo nivel.

• Difícilmente portable, es decir, un código escrito para un microprocesador, puede necesitar ser modificado, para poder ser usado en otra máquina distinta. Al cambiar a una máquina con arquitectura diferente, generalmente es necesario reescribirlo completamente.

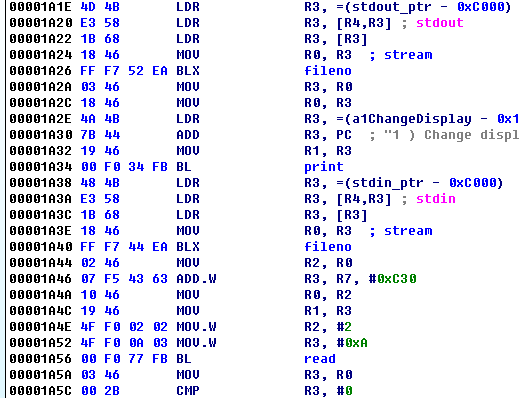
• Los programas hechos por un programador experto en lenguaje ensamblador son generalmente mucho más rápidos y consumen menos recursos del sistema (memoria RAM y ROM) que el programa equivalente compilado desde un lenguaje de alto nivel.

• Con el lenguaje ensamblador se tiene un control muy preciso de las tareas realizadas por un microprocesador por lo que se pueden crear segmentos de código difíciles y/o muy ineficientes de programar en un lenguaje de alto nivel, ya que, entre otras cosas, en el lenguaje ensamblador se dispone de instrucciones del CPU que generalmente no están disponibles en los lenguajes de alto nivel.

• También se puede controlar el tiempo en que tarda una rutina en ejecutarse, e impedir que se interrumpa durante su ejecución.

**Ventajas**

* Como trabaja directamente con el microprocesador al ejecutar un programa, pues como este lenguaje es el más cercano a la máquina la computadora lo procesa más rápido.
* Eficiencia de tamaño: Un programa en ensamblador no ocupa mucho espacio en memoria porque no tiene que cargan librerías y demás como son los lenguajes de alto nivel.
* Flexibilidad: Es flexible porque todo lo que puede hacerse con una máquina, puede hacerse en el lenguaje ensamblador de esta máquina; los lenguajes de alto nivel tienen en una u otra forma limitantes para explotar al máximo los recursos de la máquina. O sea que en lenguaje ensamblador se pueden hacer tareas específicas que en un lenguaje de alto nivel no se pueden llevar acabo porque tienen ciertas limitantes que no se lo permite.



**Desventajas**

* Tiempo de programación: Como es un lenguaje de bajo nivel requiere más instrucciones para realizar el mismo proceso, en comparación con un lenguaje de alto nivel.
* Programas fuente grandes: Por las mismas razones que aumenta el tiempo, crecen los programas fuentes; simplemente requerimos más instrucciones primitivas para describir procesos equivalentes. Esto es una desventaja porque dificulta el mantenimiento de los programas, y nuevamente reduce la productividad de los programadores.
* Peligro de afectar recursos inesperadamente: Que todo error que podamos cometer, o todo riesgo que podamos tener, podemos afectar los recursos de la máquina, programar en este lenguaje lo más común que pueda pasar es que la máquina se bloquee o se reinicie.
* Falta de portabilidad: Porque para cada máquina existe un lenguaje ensamblador; por ello, evidentemente no es una selección apropiada de lenguaje cuando deseamos codificar en una máquina y luego llevar los programas a otros sistemas operativos o modelos de computadoras.

**5. Diferencia entre Little endian y big endian**

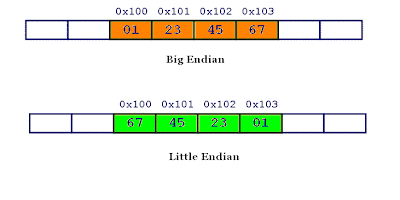
Dependiendo del sistema operativo en que se trabaje, hay que considerar el orden de los bytes en los tipos de datos numéricos que utilizan varios bytes. Existen dos formatos diferentes, denominados Little Endian y Big Endian.

1. Big-Endian: es el orden intuitivo. El primer byte guardado representa la cifra más significativa.
2. Little-Endian: es al revés de lo que uno esperaría. El primer byte guardado representa la cifra menos significativa.

¿Suena confuso? La verdad es que no lo es tanto, es solo cosa de costumbre. Si tenemos el siguiente array |x01|0x02|0x03|0x04|, entonces el número correspondiente es :

1. Big-endian: 0x01020304= 16909060
2. Little-endian: 0x04030201= 67305985

Hay que tener en cuenta que el orden físico de los bytes no cambia, es decir, que, en la memoria, el primer byte es 0x01, el segundo es 0x02 y así sucesivamente. Lo único que cambia es nuestra manera de interpretar la información.

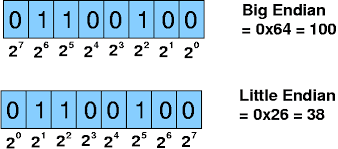


Lo que es realmente importante, es saber qué convención estamos utilizando. En general, en los protocolos de redes (TCP, UDP) se usa el formato Big-endian. En cambio, en los procesadores se suele usar Little-endian.

Los formatos "Little Endian" y "Big Endian" pueden aplicarse, además de a la ordenación de los bytes, a la ordenación de los bits.

En un byte, un sistema que utilice "Little Endian" tendrá el bit de menor peso en el primer bit y el bit de mayor peso en el último bit. En un sistema "Big Endian", el bit de mayor peso estará en el primer bit y el de menor peso en el último.

Al aplicarse tanto a la ordenación de bytes como de bits, aprovechando las ventajas de cada uno, podemos encontrar sistemas en los que se utiliza el formato "Big Endian" para los bytes y el "Little Endian" para la ordenación de los bits internamente en cada byte, o viceversa.



## 2.2 Entregas Individuales

**No aplica en esta unidad**

# 3. Programas

**Programas**

## 3.1 Entregas en Equipo

**Enlace al Repositorio:** [Ensamblador-Microprocesadores/PM\_U1\_EQ\_1 at main · emmaprofemx/Ensamblador-Microprocesadores (github.com)](https://github.com/emmaprofemx/Ensamblador-Microprocesadores/tree/main/PM_U1_EQ_1)

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nombre del Programa** | **Estado** | **Ubicación** |
| 1 | Plantilla | ENTREGADO | El programa se encuentra en la carpeta PM\_U1\_EQ\_1 |
| 2 | Plantilla Variables | ENTREGADO | El programa se encuentra en la carpeta PM\_U1\_EQ\_1 |
| 3 | Registros | ENTREGADO | El programa se encuentra en la carpeta PM\_U1\_EQ\_1 |

## 3.2 Entregas Individuales

**Enlace al Repositorio: NO APLICA EN ESTA UNIDAD**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nombre del Programa** | **Estado** | **Ubicación** |
|  |  | {ENTREGADO,  INCOMPLETO,  NO ENTREGADO} |  |

# 4. Prácticas

**Prácticas**

## 4.1 Entregas en Equipo

**Enlace al Repositorio:** [Practica de la Unidad 11 Video.mp4](https://alumnosuatedu-my.sharepoint.com/:v:/r/personal/a2163216015_alumnos_uat_edu_mx/Documents/Practica%20Unidad%201%20microprocesadores/Practica%20de%20la%20Unidad%201%20Video.mp4?csf=1&web=1&e=px2L4T)

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nombre de la Práctica** | **Estado** | **Ubicación** |
| 1 | CREACION, EXPLICACION , Y EJECUCION DE UN PROYECTO ENSAMBLADOR EJECUTANDOLO EN VISUAL STUDIO 2019. | ENTREGADO. | EL LINK DEL VIDEO SE ENCUENTRA EN EL APARTADO DE PRACTICAS. |

### 4.1.1 Desarrollo de las Prácticas

**Práctica 1. CREACION DE UN PROYECTO DE ENSAMBLADOR USANDO VISUAL STUDIO (VIDEO TUTORIAL).**

**Descripción**:

Se tiene que crear un video explicando cómo debería de crearse correctamente un proyecto en ensamblador usando visual studio 2019.

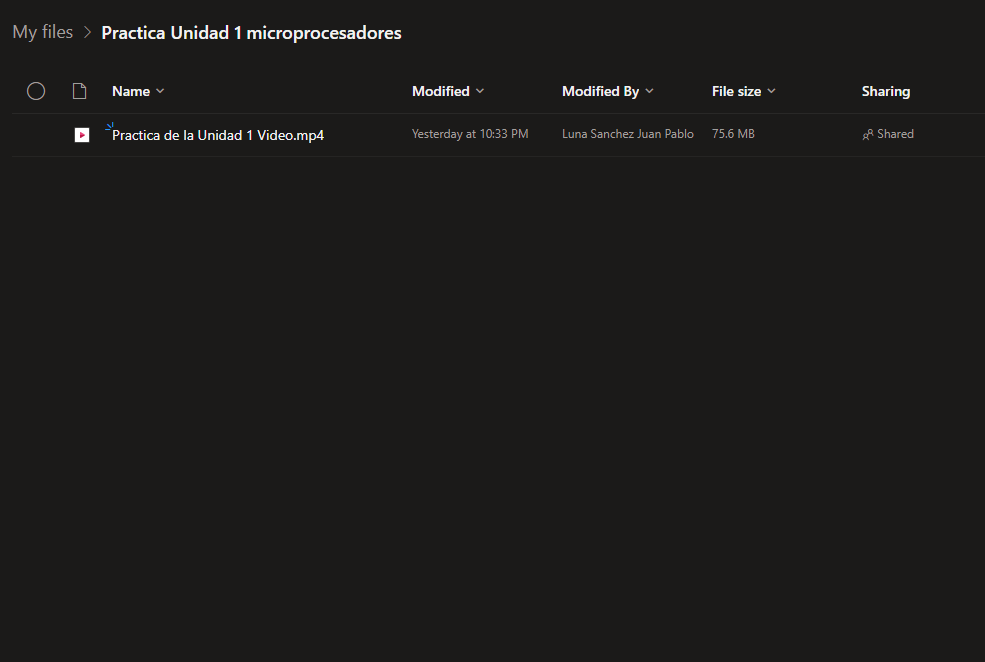
**Introducción**:

En el video se menciona desde la creación de un proyecto y configurando en un ambiente para programar en lenguaje ensamblador en visual studio 2019 hasta finalizar corriendo el programa.

**Desarrollo:**

Se fueron describiendo paso a paso en el video las configuraciones que uno como usuario tiene que hacer para que pueda trabajar en un entorno en ensamblador.

**Demostración:**



Se colocó en la plataforma One drive y el link se encuentra posicionado en el enlace al repositorio del apartado practicas.

**Conclusiones:**

Para concluir fue de gran ayuda saber cómo es que se crea y configura un proyecto para poder trabajar en un entorno de lenguaje ensamblador de esta forma nos ayudará a tener más conocimientos sobre el tema y también sabremos que este proceso se repite cada que creemos un nuevo proyecto por lo tanto nos servirá para los futuros proyectos que estemos realizando y ya sabremos como configurarlo de una manera mas rápida.

## 4.2 Entregas Individuales

**Enlace al Repositorio: NO APLICA EN ESTA UNIDAD.**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nombre de la Práctica** | **Estado** | **Ubicación** |
|  |  | {ENTREGADO,  INCOMPLETO,  NO ENTREGADO} |  |

# 5. Proyecto

**Proyecto**

**Descripción de las ventanas que se están explicando en clase**

**Enlace al Repositorio: NO APLICA EN ESTA UNIDAD.**

**Descripción**:

En este proyecto se explicará con fundamentos teóricos e imágenes los diferentes apartados como lo son las ventanas de inspector, memoria y registros que se encuentran en visual studio 2019 y que estuvimos ocupando para la realización de prácticas y ejercicios.

**Introducción**:

El lenguaje ensamblador es un lenguaje de programación de bajo nivel. Ayuda a comprender el lenguaje de programación para el código de máquina. En la computadora, hay un ensamblador que ayuda a convertir el código de ensamblaje en código ejecutable de máquina.

El lenguaje ensamblador está diseñado para comprender las instrucciones y proporcionar lenguaje de máquina para su posterior procesamiento. Depende principalmente de la arquitectura del sistema, ya sea el sistema operativo o la arquitectura de la computadora.

**Desarrollo:**

El **inspector** de Visual Studio es una herramienta en la cual nos permite visualizar las propiedades de una o más variables, al momento en que agregamos una variable dentro del inspector observamos las siguientes propiedades:

* Nombre
* Valor
* Tipo de dato

En el apartado de demostración se anexa captura de pantalla de la ventana de inspector.

La Ventana **Memoria** muestra el espacio de memoria que usa la aplicación. Nos sirve para para observar cómo se comporta la memoria al momento del modo de depuración, esta vista de memoria es conveniente para examinar grandes conjuntos de datos, que no se muestran adecuadamente en las demás ventanas.

La ventana Memoria no se limita a mostrar datos. Muestra todo lo que hay en el espacio de memoria, ya sean datos, código o bits aleatorios de la memoria no asignada.

Para acceder a la ventana de Memoria se siguen los siguientes pasos en el apartado de demostración memoria.:

Los **registros** del procesador son una porción de memoria ultrarrápida, se emplean para:

• Controlar instrucciones en ejecución.

• Manejar direccionamiento de memoria.

• Proporcionar capacidad aritmética.

Los registros son espacios físicos dentro del microprocesador con capacidad de 4 bits hasta 64 bits dependiendo del microprocesador que se emplee.

• Los bits, por conveniencia, se numeran de derecha a izquierda: (15,14,13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0).

Estos son los únicos registros que pueden usarse de modo dual (en 8 o 16 bits)

Los registros de la CPU son conocidos por sus nombres propios, que son: ·

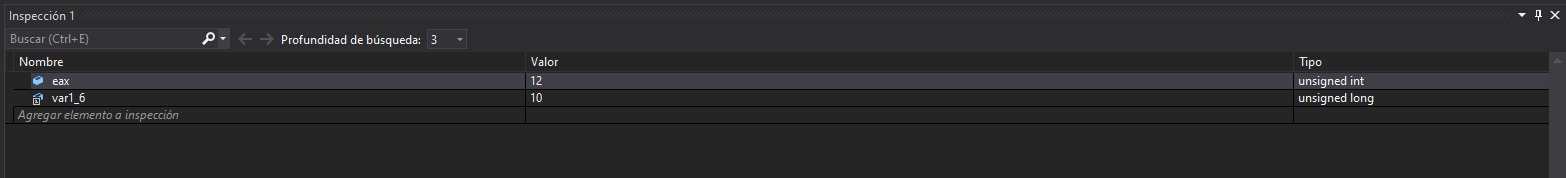
* AX (acumulador).
* BX (registro base).
* CX (registro contador).
* DX (registro de datos).

En el apartado de demostración se muestran capturas de la ventana de registro.

**Demostración:**

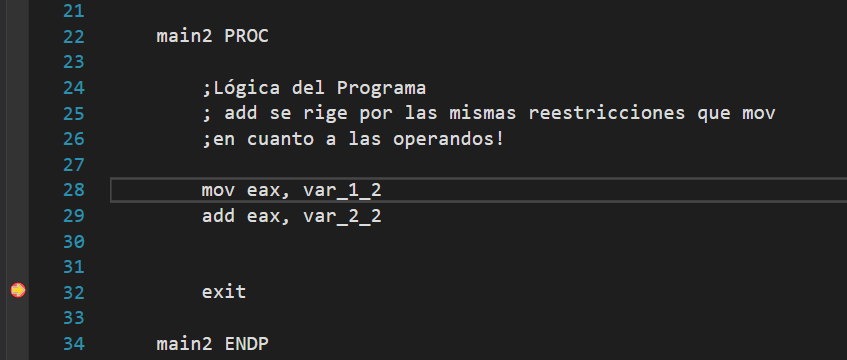
**Inspector**

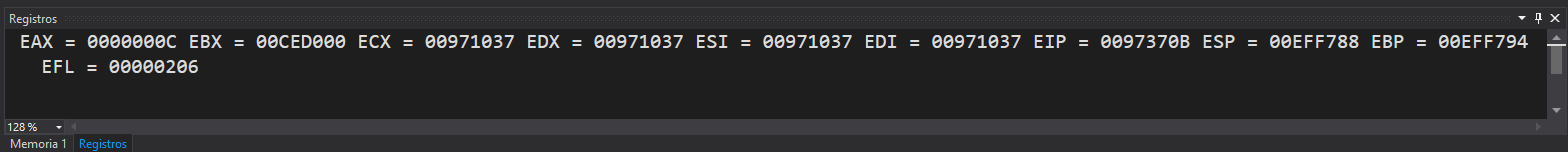
Captura de inspector donde se visualiza el nombre, el valor y el tipo de dato

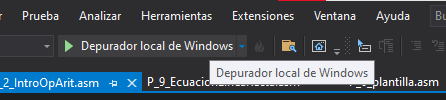


Memoria capturas de pantalla

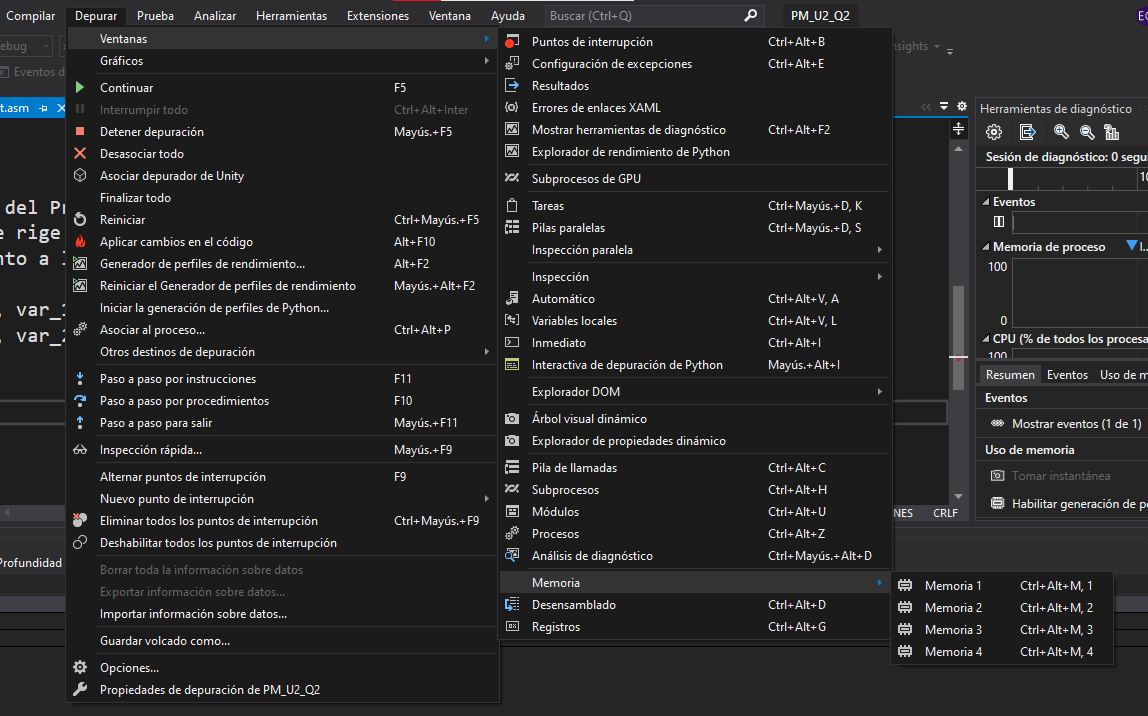
1. Primero, agrega un breakpoint en el cual harás la depuración



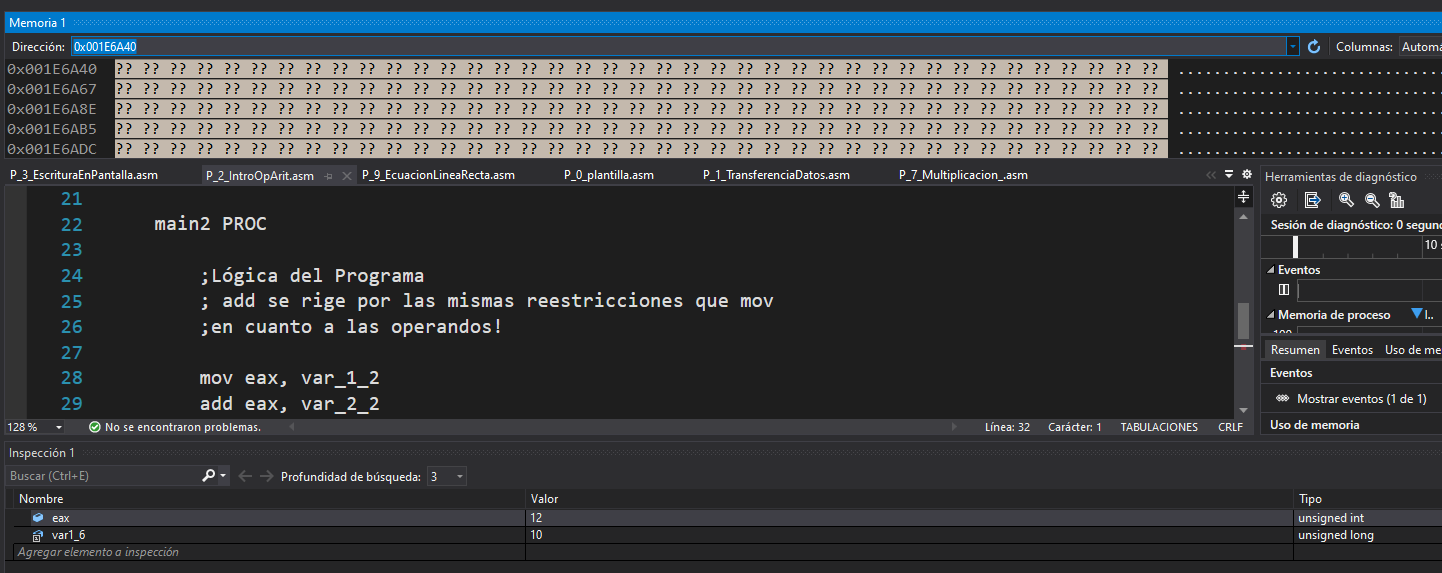
1. En seguida, procede a ejecutar el programa.



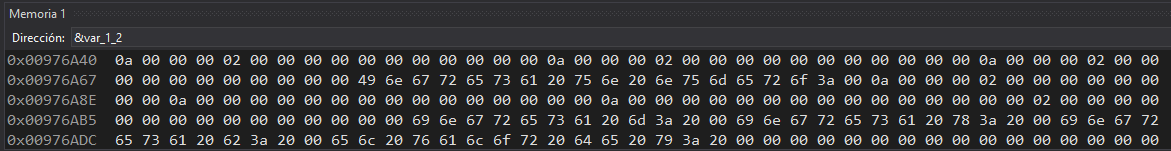
1. Ya depurando el programa, nos vamos a la barra de herramientas en la opción de Depurar->Ventanas->Memoria->Memoria 1.



1. En seguida observamos la ventana de la memoria



En seguida observamos la ventana de la Memoria, para observar el comportamiento de una variable se escribe en el apartado de dirección se sigue la siguiente regla &nombre\_variable , como se muestra en el siguiente:



De esta manera observamos cómo se comporta una variable internamente cuando depuramos nuestro programa.

**Conclusiones:**

Es interesante ver como cada ventana nos permite visualizar lo que estamos haciendo caso del inspector que las variables que vayamos agregando podremos visualizarlas a través del inspector además permite seleccionar elementos puntuales para ver y modificar su contenido: registros, posiciones de memoria, variables, etc. , así como la ventana de memoria el cual mostrara el contenido de cualquier zona de la memoria virtual del programa en formato hexadecimal de 8 bits (byte), 16 (Short Hex) o 32 (Long hex). El programador puede modificar el contenido de la memoria que aparece en la ventana. Y por último podemos entender que la ventana registro contenera los registros que se pueden modificar desde una aplicación de usuario.