Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Laboratorio Arquitectura de Computadoras y Ensambladores 1
Sección “N”
Auxiliar: Robinson Pérez

MANUAL TÉCNICO
PROYECTO 2 FASE 1

Nombre y Apellidos
Pedro Antonio Castro Calderón

Carné
201900612

Guatemala 24 de Diciembre de 2022

DESCRIPCIÓN DE LA APLICACIÓN

La aplicación consiste en un programa creado en lenguaje ensamblador x86, que será capaz de recibir una función de grado n (no mayor a 5) , para posteriormente poder visualizarla en pantalla, resolver su respectiva derivada e integral con la opción de imprimirla también.

HERRAMIENTAS UTILIZADAS

Sistema Operativo: Elementary OS 6.1 Jólnir

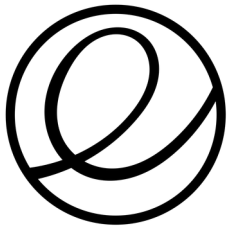
RAM: 8GB

IDE: Visual Studio Code 1.74.1

Ensamblador Utilizado: NASM x86

Consola: Terminal de Linux

Emulador: DOSBox 0.74 -3



ESTRUCTURA DEL PROYECTO

Todo el código está en un solo archivo, llamado Entrada.asm, utilizando la sintaxis de Intel con NASM x86.

En el código, se utilizaron los siguientes registros:

- Registros de 8 bits : al, ah, bl, bh, cl, ch, dl, dh
- Registros de 16 bits: ax, bx, cx, dx

Interrupciones:

Se utilizaron interrupciones de DOS (int 21h) como por ejemplo

- 09h : Imprimir en pantalla
- 01h: Leer un carácter ingresado por el usuario y desplegarlo
- 02h: Despliega un carácter en la pantalla

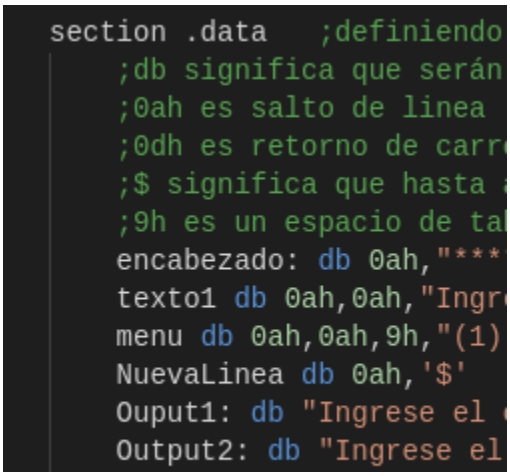
Se utilizaron interrupciones de BIOS (int 10h) como por ejemplo

- 00h: Poner el modo video
- 03h Obtener la posición y el tamaño del cursor

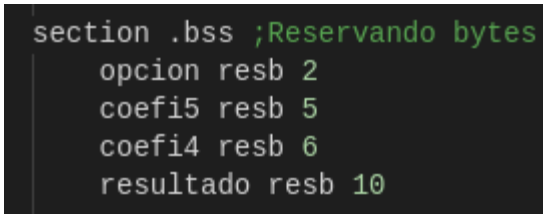
Secciones del código

El código está dividido en 3 secciones:

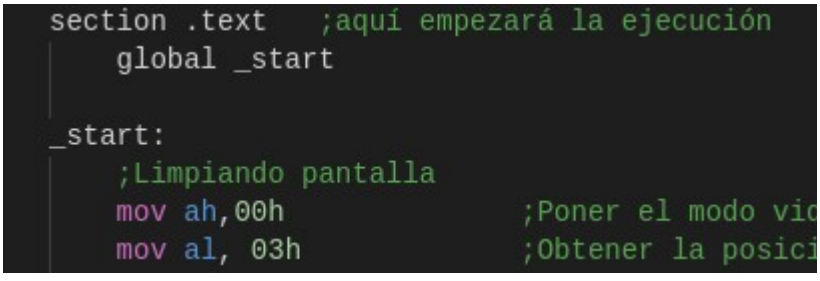
- **section .data:** Esta sección se usa para declarar datos inicializados o constantes. Aquí se pueden declarar valores, nombres de archivo, tamaños de buffer, etc.



- **section .bss:** La sección bss es utilizada para declarar variables, para usarlos en el futuro durante la ejecucioń del programa.



- **section .text:** En esta sección va el código como tal, Esta sección debe empezar con la declaración *global _start* , lo cuál, le dice al kernel dónde comienza la ejecución del programa.



Comentarios

Los comentarios en el lenguaje ensamblador empiezan con un punto y coma (;) , en el código fuente se hizo uso de comentarios para especificar las acciones del código, por ejemplo:

; Esta interrupción muestra un mensaje en la pantalla

Procedimientos:

Almacenan código y son llamados con la palabra reservada *call* , para la creación de la aplicación, se crearon los siguientes procedimientos

- **_getEntrada:** Recibe la opción de menú dónde el usuario desea dirigirse, si existe, lo redirigirá a esa opción.
- **_ingresarCoeficientes:** Procedimiento para ingresar los coeficientes de la ecuación
- **_verFuncion:** Despliega en pantalla la última función almacenada en memoria.
- **_noEsNumero:** Devuelve un mensaje de error al detectar que el usuario ingreso algo distinto a números enteros
- **_imprimirDerivada:** Resuelve la derivada de la función almacenada y la imprime al usuario.
- **_imprimirIntegral**: Resuelve la integral de la función almacenada y la imprime al usuario.
- **_noDisponible:** Muestra un mensaje de que la opción que el usuario ingresó aún no está disponible en la aplicación.
- **_lineaNueva:** Imprime una nueva linea
- **_Salir:** Sale del programa mostrando un mensaje previo a finalizar la ejecución
- **_Escape:** Interrumpe la ejecución del programa.
- **_opcionInvalida:** Muestra un mensaje de que la opción que el usuario ingresó no es válida.