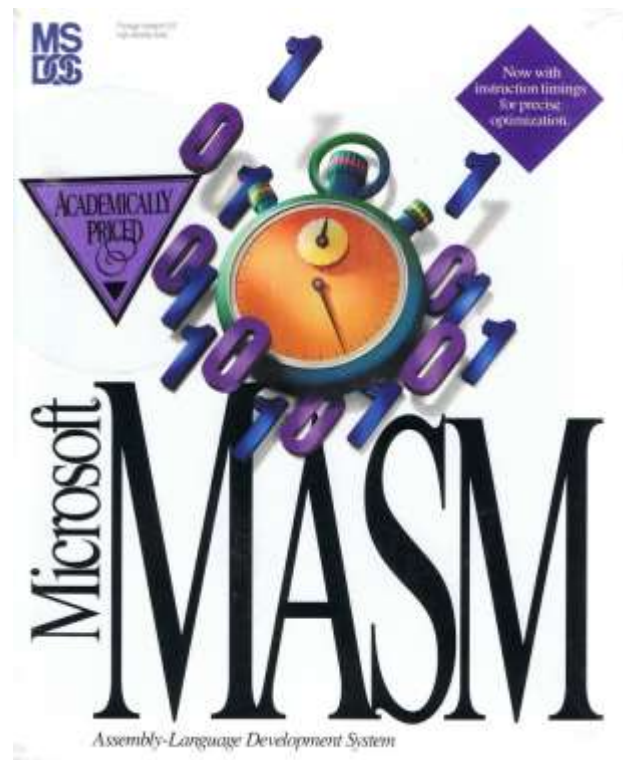


# Proyecto 1

## (Manual De Programador)



## Contenido

Objetivos y alcances del sistema .....	3
Especificaciones Técnicas .....	3
Lógica del programa .....	4
Instalación de aplicación .....	6

## Objetivos y alcances del sistema.

### General:

Brindar al lector una guía que contenga la información del manejo de variables, métodos y macros para futuras actualizaciones y modificaciones realizadas por terceros.

### Específicos:

- Mostrar al lector una descripción lo más completa y detallada posible del SO, IDE entre otros utilizados para el desarrollo de la aplicación.
- Proporcionar al lector una concepción y explicación técnica - formal de los procesos y relaciones entre métodos y atributos que conforman la parte operativa de la aplicación.

## Especificaciones Técnicas

Sistema Operativo: Windows.

Lenguaje de Programación: MASM.

IDE: Visual Studio Code.

Lógica del programa.

**Archivos utilizados:**

- **main:** Es la clase principal donde se inicializa el programa.
- **macros:** Es la clase complementaria, que contienen todos los procedimientos del menú y sus funciones.

**Librerías utilizadas:**

No se utilizó ninguna clase de librería, todo fue con métodos nativos de MASM

## Flujo del programa:

- ◆ El programa funciona de la siguiente manera:

Al iniciar el programa se ejecuta el archivo main.asm.

Llamamos el otro archivo de macros con nuestros procedimientos, `.model small` para inicializar la aplicación, llamamos el `stack` y `.data` donde iran todas nuestras variables:

```
include macros.asm; copia lo del archivo de macros
;-----
.model small; Encargado de establecer el tamaño de memoria que usaremos, en este caso small.

.stack; Proporciona al procesador el tamaño de la pila.

.data; Segmento donde estableces los espacios de memoria y las variables a utilizar.
; variables auxiliares para las multiplicaciones y divisiones
```

```
; variables auxiliares para las multiplicaciones y divisiones
numerrr dw 2, '$'
numero dw 0010h, '$'
num2 dw 2, '$'
num3 dw ?, '$', '$', '$', '$', '$', '$'
expo6 db 0006h, '$'
expo5 db 0005h, '$'
expo4 db 0004h, '$'
expo3 db 0003h, '$'
expo2 db 0002h, '$'
; Cadenas que me sirven para el menu
menuTexto db 0ah, 0dh, '-----Menu Proyecto Unico-----:', '$'
cadena db 0ah, 0dh, 'Escoje una opcion:', '$'
cadenaDos db 0ah, 0dh, '1) Ingresa tu funcion:', '$'
cadenaTres db 0ah, 0dh, '2) Imprimir funcion almacenada:', '$'
cadenaCuatro db 0ah, 0dh, '3) Imprimir la derivada de la funcion almacenada:', '$'
cadenaCinco db 0ah, 0dh, '4) Imprimir la integral de la funcion almacenada:', '$'
cadenaSeis db 0ah, 0dh, '5) Graficar:', '$'
cadenaSiete db 0ah, 0dh, '6) Metodo de Newton:', '$'
cadenaOcho db 0ah, 0dh, '7) Metodo de Steffensen:', '$'
cadenaNueve db 0ah, 0dh, '8) Salir de la aplicacion:', '$'
cadenaExtra db 0ah, 0dh, 'Introduce tu numero deseado:', '$'

cadenaCoCinco db 0ah, 0dh, 'Introduce tu coeficiente para el exponente 5: ', '$'
cadenaCoCuatro db 0ah, 0dh, 'Introduce tu coeficiente para el exponente 4: ', '$'
cadenaCoTres db 0ah, 0dh, 'Introduce tu coeficiente para el exponente 3: ', '$'
cadenaCoDos db 0ah, 0dh, 'Introduce tu coeficiente para el exponente 2: ', '$'
cadenaCoUno db 0ah, 0dh, 'Introduce tu coeficiente para el exponente 1: ', '$'
cadenaCoCero db 0ah, 0dh, 'Introduce tu coeficiente para el termino constante: ', '$'

backMenu db 0ah, 0dh, 'Presiona cualquier tecla para volver al Menu:', '$'
despedida db 0ah, 0dh, 'Gracias por usar mi program!!', '$'
aceptado db 0ah, 0dh, 'Tu ecuacion fue almacenada', 10, 13, '$'
respuesta db 0ah, 0dh, 'Esta es tu ecuacion almacenada: ', 10, 13, '$'
respuesta2 db 0ah, 0dh, 'La derivada de tu ecuacion es: ', 10, 13, '$'
respuesta3 db 0ah, 0dh, 'La integral de tu ecuacion es: ', 10, 13, '$'
```

En la sección de code llamamos a nuestra macro de menú que nos mantiene en loop el menú con sus opciones, si elige alguna que no está en las opciones predefinidas te volverá a dejar elegir:

```
.code;Segmento de todo el codigo o bien el conocido main.
main PROC;Inicio de mi clase main.
    mov dx, @DATA
    mov ds, dx
    menu;Mi metodo menu principal
main ENDP;Fin de mi clase main.
```

```
menu MACRO;macro que mantiene el menu y lo repite en loop.
menuPrincipal;;imprimo todas las opciones del menu
    textos
    leerchar;leo el caracter que introducen
    cmp al, 31h;compruebo si el caracter es igual a 31h que es el codigo hexadecimal del numero 1 en ascii
    je opcionUno;uso el jmp equals para ver si es igual y si si, pues entro en la etiqueta
    cmp al, 32h
    je opcionDos
    cmp al, 33h
    je opcionTres
    cmp al, 34h
    je opcionCuatro
    cmp al, 35h
    je opcionCinco
    cmp al, 36h
    je opcionSeis
    cmp al, 37h
    je opcionSiete
    cmp al, 38h;si eligen 8, salgo del programa
    je salir
    jmp menuPrincipal;vuelvo a poner el menu principal si ponen algun caracter que no es los que considere
opcionUno;;etiquetas de las opciones, me manda a su macro especial
    trabajoUno
    print backMenu
    leerchar;al terminar el trabajo de la macro debo dar alguna tecla para volver al menu
    limpiar
    jmp menuPrincipal
```

Este trabaja a base de comparaciones y te manda a su bandera especifica para realizar su macro individual, ya sea ingresar los datos, imprimir la ecuación guardada, imprimir la derivada de la función o imprimir la integral.

```

trabajoUno MACRO;macro de ingresar ecuacion
    limpiar
    print cadenaCoCinco
    leerTexto coeCinco;almacena cada coeficiente en su variable

    limpiar
    print cadenaCoCuatro
    leerTexto coeCuatro

    limpiar
    print cadenaCoTres
    leerTexto coeTres

    limpiar
    print cadenaCoDos
    leerTexto coeDos

    limpiar
    print cadenaCoUno
    leerTexto coeUno

    limpiar
    print cadenaCoCero
    leerTexto coeCero

    limpiar
    print aceptado
ENDM

```

```

trabajosDos MACRO;Esta macro imprime la ecuacion almacenada, solo si son una variable, si son cero no los imprime
    limpiar
    comparaCero coeCinco, coeCuatro, coeTres, coeDos, coeUno, coeCero
ENDM
;
trabajoTres MACRO;Esta macro imprime la derivada de la ecuacion almacenada, solo si son una variable, si son cero no los imprime
    limpiar
    comparaUno coeCinco, coeCuatro, coeTres, coeDos, coeUno, coeCero
ENDM
;
trabajoCuatro MACRO;Esta macro imprime la integral de la ecuacion almacenada, solo si son una variable, si son cero no los imprime
    limpiar
    comparaDos coeCinco, coeCuatro, coeTres, coeDos, coeUno, coeCero
ENDM
;

```

Repositorio del proyecto.

Todo el código se encuentra comentado.

[https://github.com/stvnrr/PROYECTO\\_UNICO\\_ARQUI1\\_DIC-2022.git](https://github.com/stvnrr/PROYECTO_UNICO_ARQUI1_DIC-2022.git)