

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Arquitectura de computadores y ensambladores 1
Ing. Otto Escobar
Auxiliar sección A: David Jonathan González Gámez.
Auxiliar sección A: Andhy Lizandro Solís Osorio.
Auxiliar sección B: José Fernando Valdéz Pérez.
Auxiliar sección B: María de Los Angeles Herrera Sumalé.



Proyecto 2

Objetivo General:

- Aplicar los conocimientos adquiridos en el curso sobre el lenguaje ensamblador y arduino.

Objetivos Específicos:

- Aplicar el conocimiento de operaciones básicas a nivel ensamblador.
- Utilizar procedimientos y macros
- Manipular la memoria a bajo nivel
- Realizar operaciones aritméticas a bajo nivel
- Hacer uso de interrupciones básicas para interactuar con el sistema operativo
- Hacer uso de comunicación serial con ensamblador y arduino

Descripción

Se pide que desarrollen un programa con ensamblador para realizar los siguientes cálculos con funciones: derivar, integrar, resolver y graficar. También se debe de comunicar ensamblador con arduino por medio de comunicación serie. En arduino se podrá graficar una función enviada desde ensamblador.

El coeficiente en la función puede ser un número de hasta dos dígitos y este puede ser positivo o negativo. (-99 a 99)

Archivo de entrada

Las funciones en el archivo de entrada se escribirán separándolas por punto y coma y el salto de línea. No hay un máximo de funciones a ingresar por archivo, pero en el sistema podrán existir como máximo 20 funciones, ya sean ingresadas a mano, por archivo, las funciones ya derivadas e integradas.

Ejemplo:

```
x^2+5x+6;  
-3x^3+7x^2+8x-12;  
2x+1;  
8x^3-66;
```

no habrá divisiones directas en el archivo de entrada

Funcionalidades Ensamblador:

Menú principal

1. **Derivar función:** Se ingresará el id de la función de grado máximo 4 la cual únicamente tendrá coeficientes enteros. Los coeficientes serán enteros positivos y negativos solamente, se calculará la derivada de la función y debe guardarse en memoria con un identificador: d<id función original>
2. **Integrar función.** Se ingresará el id de la función de grado máximo 4 la cual únicamente tendrá coeficientes enteros. Los coeficientes serán enteros positivos y negativos solamente. Se calculará la integral de la función y debe guardarse en memoria con un identificador: i<id función original>
3. **Ingresar Funciones.** Se mostrará un menú el cual se tendrán tres opciones las cuales será:
 - a. **Ingresar Función:** Se ingresará una función de grado máximo 4 la cual únicamente tendrá coeficientes enteros. La misma deberá ser ingresada mediante una cadena la cual se analizará después. Los coeficientes serán enteros positivos y negativos solamente. La entrada también debe de

detectar que no se ingresen valores no válidos (sólo debe aceptar números enteros) y comprobar los caracteres que se están ingresando en la función. De ser correcto se calculará la derivada de la función ingresada y debe guardarse en memoria tanto la función ingresada como la derivada calculada.

- b. **Cargar Archivo:** Se ingresará la ruta de un archivo el cual contendrá un listado de funciones las cuales serán cargadas al programa y vendrán separadas por punto y coma estas funciones tendrán el mismo criterio de aceptación que las que se ingresan manualmente en el programa.

Ejemplo de archivo:

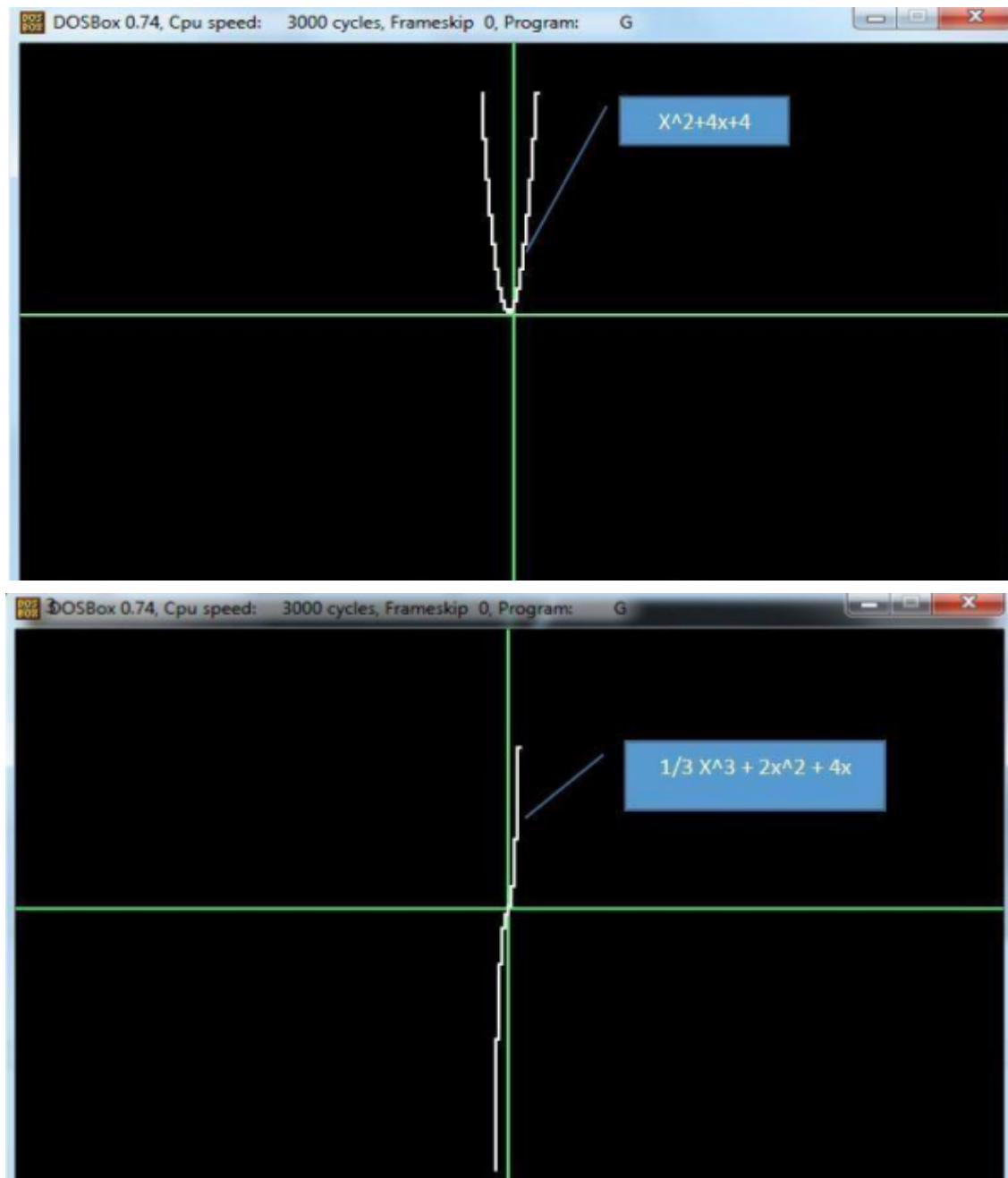
$4x^2+2x+1;$
 $-3x^4+5x+12;$

- c. **Regresar al menú principal.**

- 4. **Imprimir funciones ingresadas:** Se imprimen las funciones en memoria y se debe mostrar una advertencia si no hubiese función almacenada anteriormente.
- 5. **Graficar:** Al momento de elegir esta opción se debe comprobar que existan funciones almacenadas en el programa, de ser así se debe mostrar un mensaje que indique "Ingrese ID de la función a graficar:"

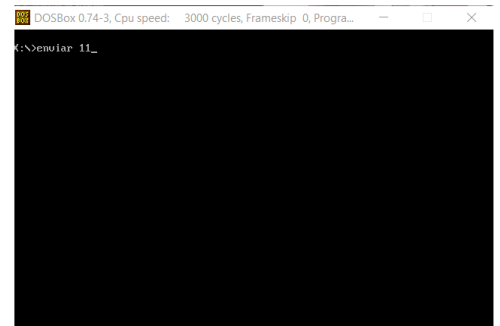
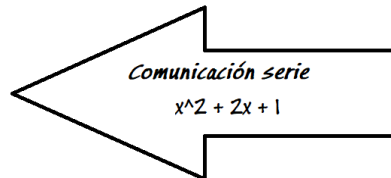
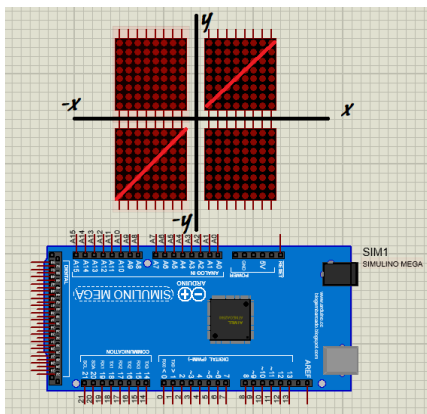
Se debe de imprimir en pantalla, utilizando el modo de video, la gráfica correspondiente al ID ingresado. No se permite el uso de otra manera de graficar la función ni librerías. La gráfica debe ser comprensible. Se debe de trazar un eje X y uno Y en los cuales se visualice las gráfica correspondiente intentando visualizar la gráfica de la mejor forma posible, el estudiante puede definir un intervalo de valores por defecto, solicitar en la entrada el intervalo a graficar.

En la parte superior se debe mostrar la función que origina la gráfica que se está mostrando. Se sugiere el siguiente modelo



6. **Resolver ecuación.** Se debe ingresar una ecuación de grado no mayor a 2 y se debe de mostrar los valores que resuelven la ecuación, de no tener solución se debe mostrar un mensaje que indique que no se puede resolver. El método de resolver la ecuación queda a criterio del estudiante.
7. **Enviar a arduino:** Pedirá el identificador de la función que se desea enviar al arduino por comunicación serie.
8. **Salir.** Terminará con la ejecución del programa

Funcionalidades Arduino:



Graficar función:

- Arduino mediante la comunicación del puerto serial, estará a la espera de que el ensamblador le envíe una función.
- Cuando Arduino recibe la función esté la analiza y la gráfica en 4 matrices led.(La función se debe de ajustar al tamaño de las 4 matrices led, a modo de que se pueda ver el comportamiento de la gráfica)
- La gráfica se muestra en la matriz durante un tiempo considerable (30seg por ejemplo) y luego se elimina
- Arduino vuelve a estar a la escucha de que el ensamblador le envíe otra función.

Entregables:

- Código fuente de Arduino y ensamblador (Archivos .asm)
- Archivo Ejecutable
- Manuales:
 - Técnico Arduino y ensamblador: Descripción del uso de los métodos y variables
 - Usuario: Funcionamiento del proyecto

Restricciones:

- Para la calificación será desde el archivo ejecutable, **no se calificará desde ningún IDE(emu 8086 o similar).**
- **Si no se manejan números negativos las funcionalidades que los utilicen se calificarán sobre el 50%.**
- **La práctica es en grupos (máximo 3) y se calificará de lo que se entregue en UEDI.**
- **No se permitirá compilar el programa por ningún motivo durante la calificación.**
- **No se permitirá utilizar ninguna librería, todo el código debe ser desarrollado por el estudiante.**
- **El programa se calificará a través de dosbox.**
- **Se debe entregar en UEDI antes de las 23:59 horas del 02 de Noviembre, el formato de entrega es [ACYE1]Proyecto2_Grupo# ejemplo:[ACYE1]Proyecto2_Grupo3.**