

Proyecto #2: “Calculadora con gráficos”.

Desarrollar un programa en lenguaje ensamblador para arquitectura Intel x86 que muestre una calculadora de manera gráfica en pantalla que debe contener los siguientes botones:

- El botón 0
 - El botón 1
 - El botón 2
 - El botón 3
 - El botón 4
 - El botón 5
 - El botón 6
 - El botón 7
 - El botón 8
 - El botón 9
 - El botón C
 - El botón <<
 - El botón + (operación Suma)
 - El botón – (operación Resta)
 - El botón / (Operación División)
 - El botón * (Operación Multiplicación)
 - El botón % (Operación Módulo [residuo de la división])
 - El botón =
-
- El programa debe funcionar con el uso del mouse.
 - El usuario debe introducir números haciendo clic en los botones numéricos del mouse.
 - Por default, el programa debe iniciar solicitando el primer número. El usuario podrá introducir hasta 4 dígitos.
 - Cuando se ha introducido el primer número, el usuario puede hacer clic en algún botón de operación (suma, resta, multiplicación, división o módulo) y el programa continuará con el segundo número.
 - El segundo número también se introducirá con los botones numéricos en pantalla y puede ser de hasta 4 dígitos.
 - Una vez que el segundo número ha sido introducido, el usuario puede hacer clic sobre el botón ‘=’ y el programa realizará el cálculo de la operación seleccionada utilizando los dos números ingresados.
 - Cuando se haya calculado, el programa debe imprimir en pantalla el resultado de la operación.
 - El usuario podrá utilizar el botón ‘C’ para limpiar el contenido en la pantalla de la calculadora. El botón ‘C’ debe reiniciar la calculadora a su estado inicial.
 - El botón ‘<<’ permite al usuario borrar carácter por carácter desde el último ingresado. Si el usuario ha dado clic en el botón ‘=’, entonces el botón ‘<<’ puede reiniciar la calculadora a su estado inicial (similar al botón ‘C’), si y sólo si se ha presionado el botón ‘=’ previamente.

Indicaciones

- Los números deben estar en sistema decimal (dígitos de 0-9).
- Los números deberán ser enteros sin signo.
- Cada número introducido por el usuario puede ser de hasta 4 dígitos. El programa debe restringir que el usuario introduzca más números.

Qué debe hacer el programa:

Al ejecutar el programa, se deberá solicitar un número e ingresarlo con ayuda de los botones numéricos en pantalla. El programa debe detectar el movimiento del mouse y el clic izquierdo. El programa debe detectar el clic dentro de los límites de cada botón para que el clic sea válido. Si el clic se hace fuera de los botones, ninguna acción debe ocurrir en el programa.

Cada número ingresado debe estar en formato decimal y puede ser de hasta 4 dígitos. El programa no deberá avanzar si no se ha introducido al menos un dígito y sólo permitirá se introduzcan hasta 4.

Una vez introducido el primer número, al dar clic en algún botón de operación, el programa continuará y solicitará un segundo número.

Cuando se han introducido ambos números, el usuario podrá hacer clic sobre el botón '=' y el programa calculará el resultado de la operación entre ambos números. El resultado se deberá imprimir en pantalla.

El usuario puede hacer clic sobre el botón 'C' en cualquier momento, para limpiar la pantalla de la calculadora.

El usuario puede hacer clic sobre el botón '<<' para retroceder un carácter desde el último ingresado y lo puede hacer hasta el inicio. Sólo si el resultado se muestra en pantalla, es posible que este botón tenga el mismo funcionamiento que el botón 'C'.

La ejecución del programa debe ser continua hasta que el usuario decida cerrar el programa haciendo clic en un botón específico para ello (función del botón 'X' [en la esquina superior derecha de la pantalla], agregado en el código base).

Las operaciones se deben calcular de la siguiente manera:

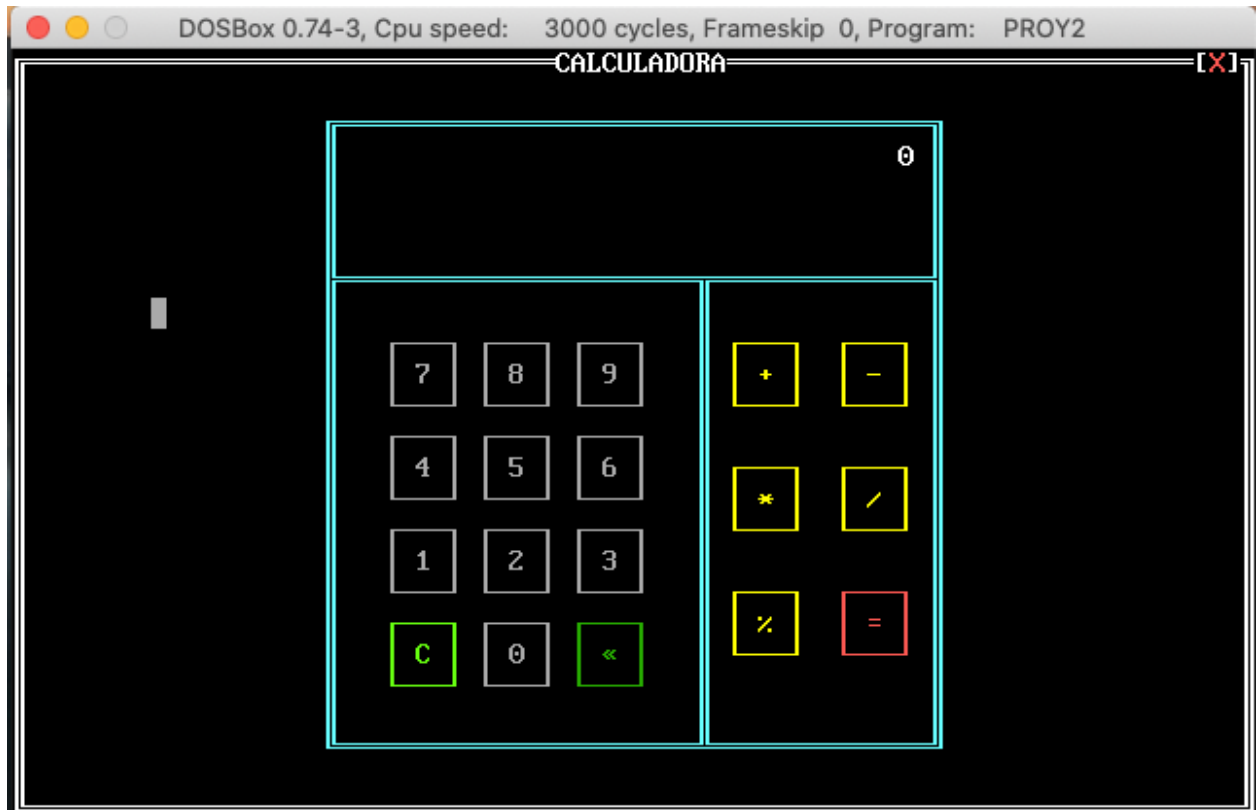
- Suma: el primer número más el segundo.
- Resta: el primer número menor el segundo.
- Multiplicación: el primer número por el segundo.
- División (cociente): el primer número entre el segundo.
- Módulo (residuo): el primer número entre el segundo.

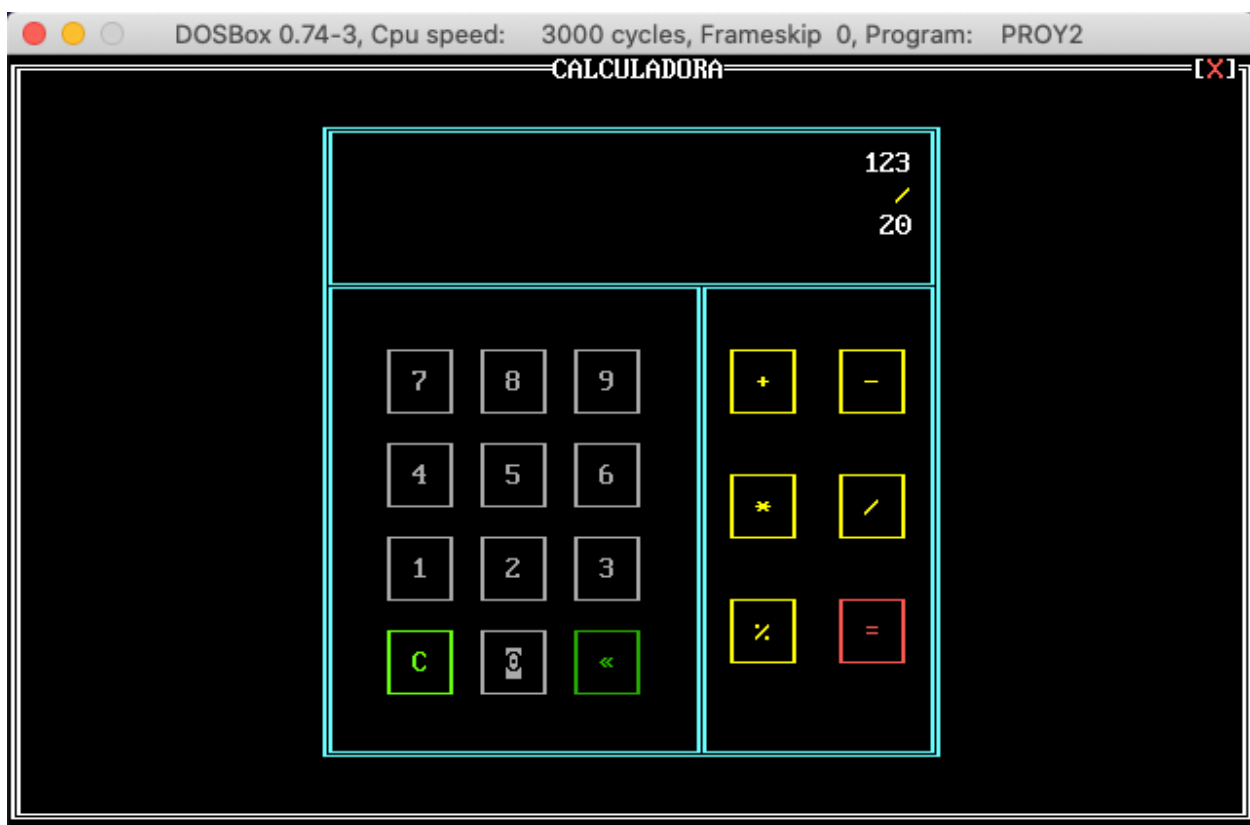
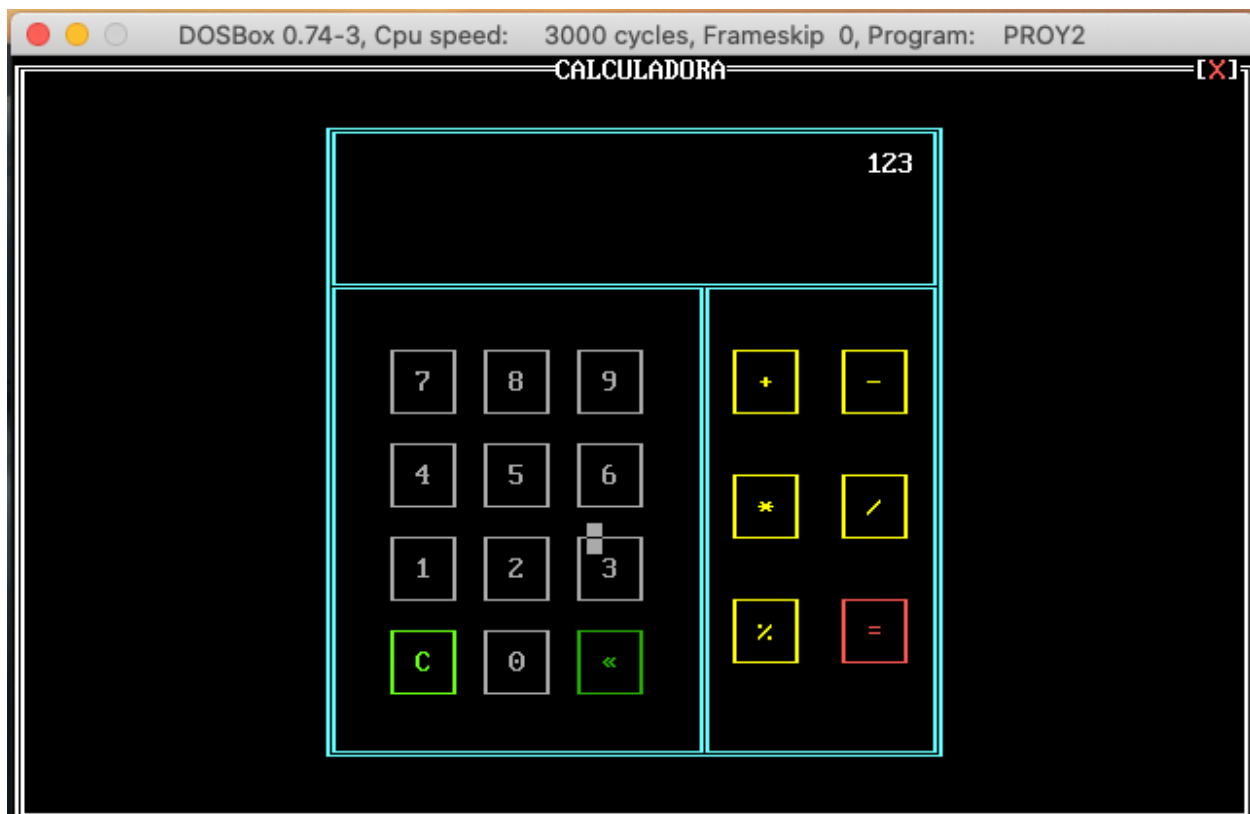
Cuando la resta da como resultado un número negativo, se debe de mostrar ese resultado, acompañado del carácter '-'. Por ejemplo, si el primer número ingresado es 1 y el segundo es 2, el resultado mostrado en pantalla debe ser -1, o equivalente (-01, -001, -0001, -00001, etc.)

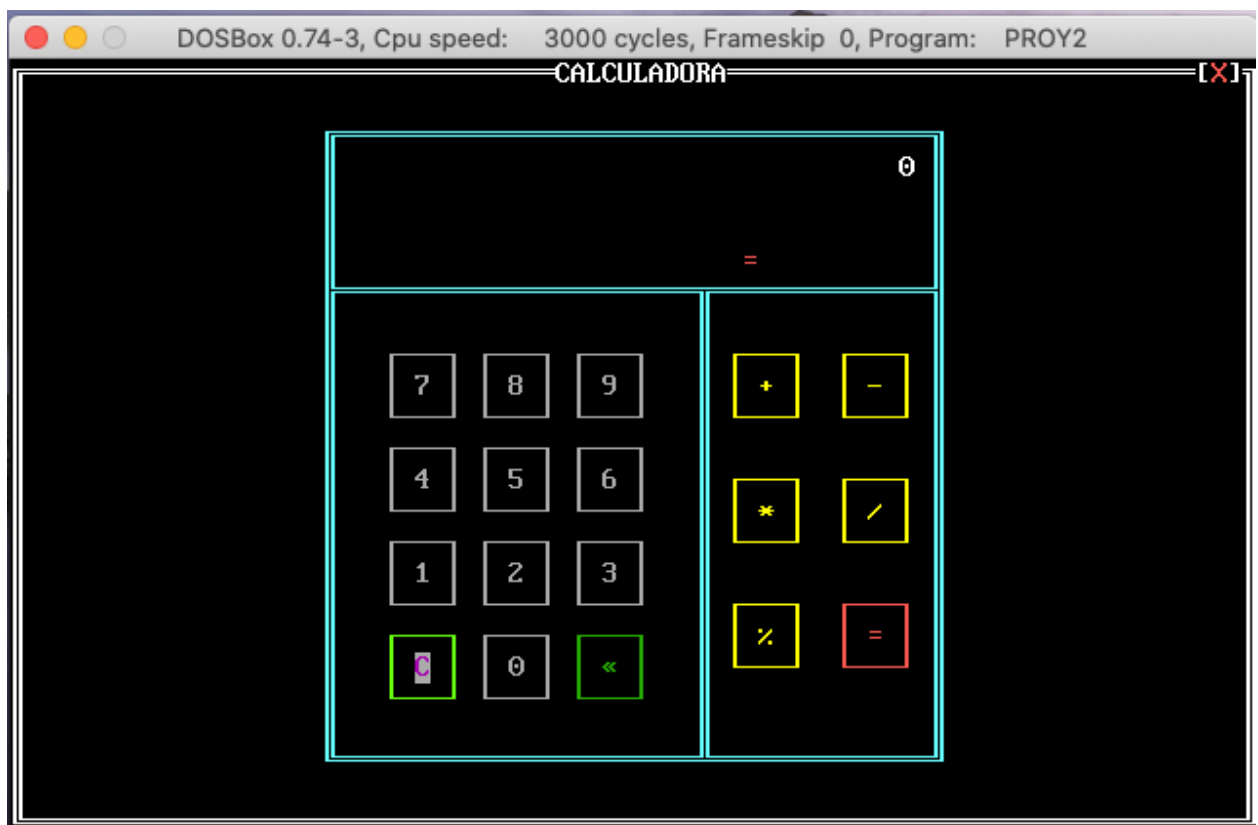
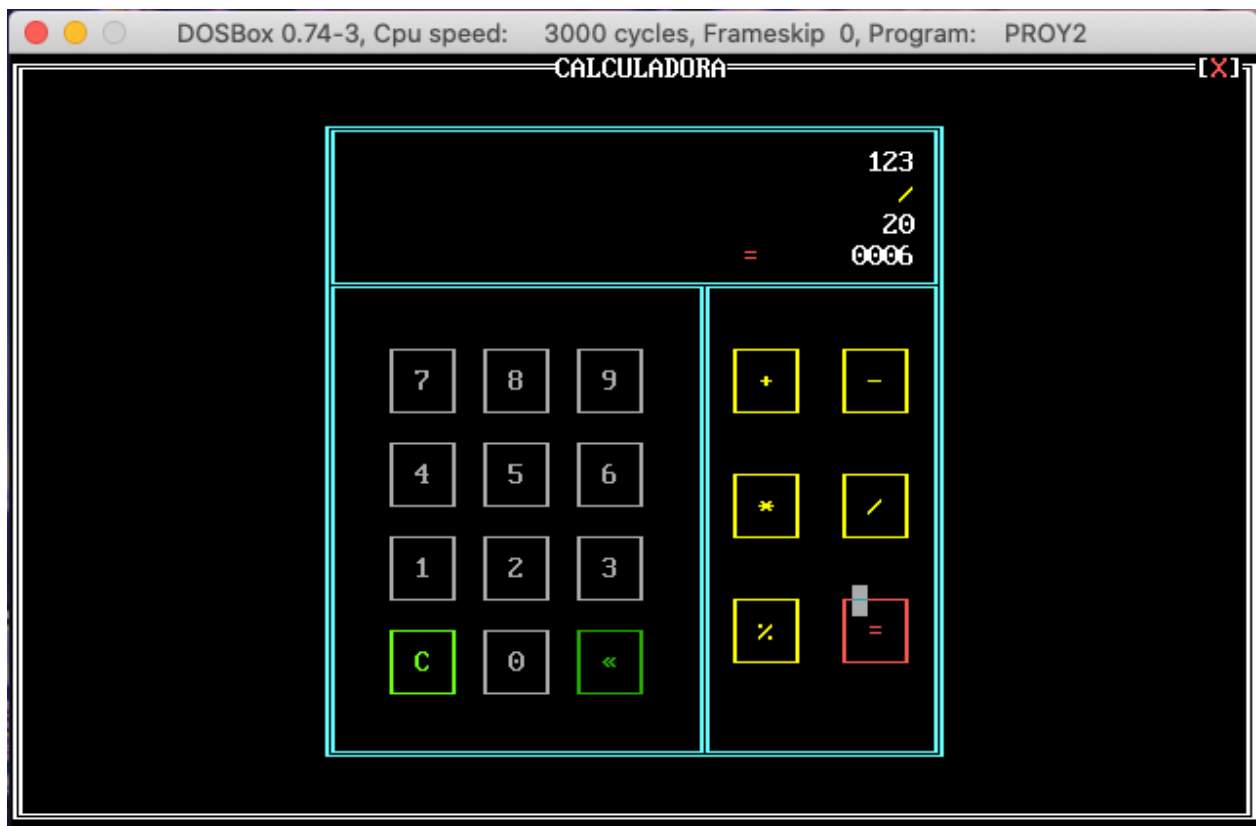
Cuando la división o el módulo se hace con un divisor igual a 0, el programa no debe detenerse y sólo especificar que el resultado no se puede obtener (puede ser cualquier leyenda, como: "Err", "Error!",

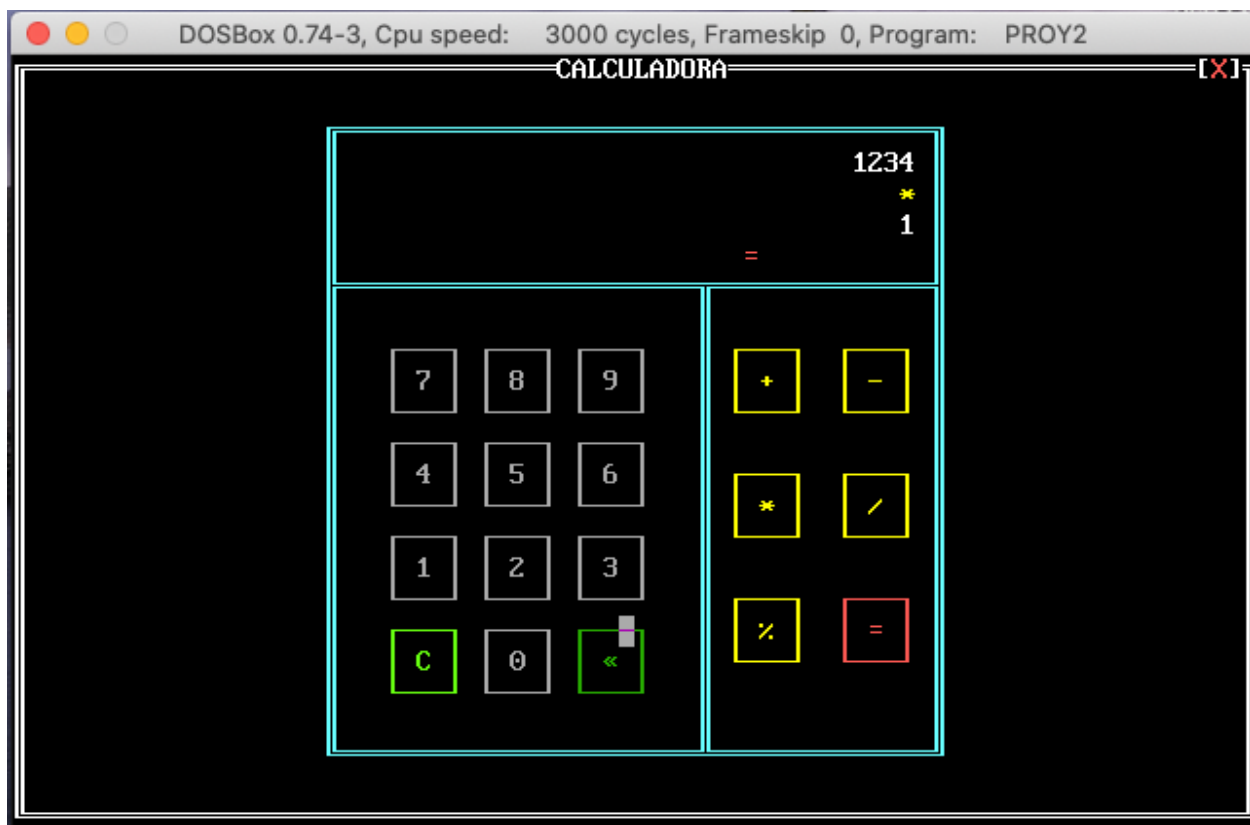
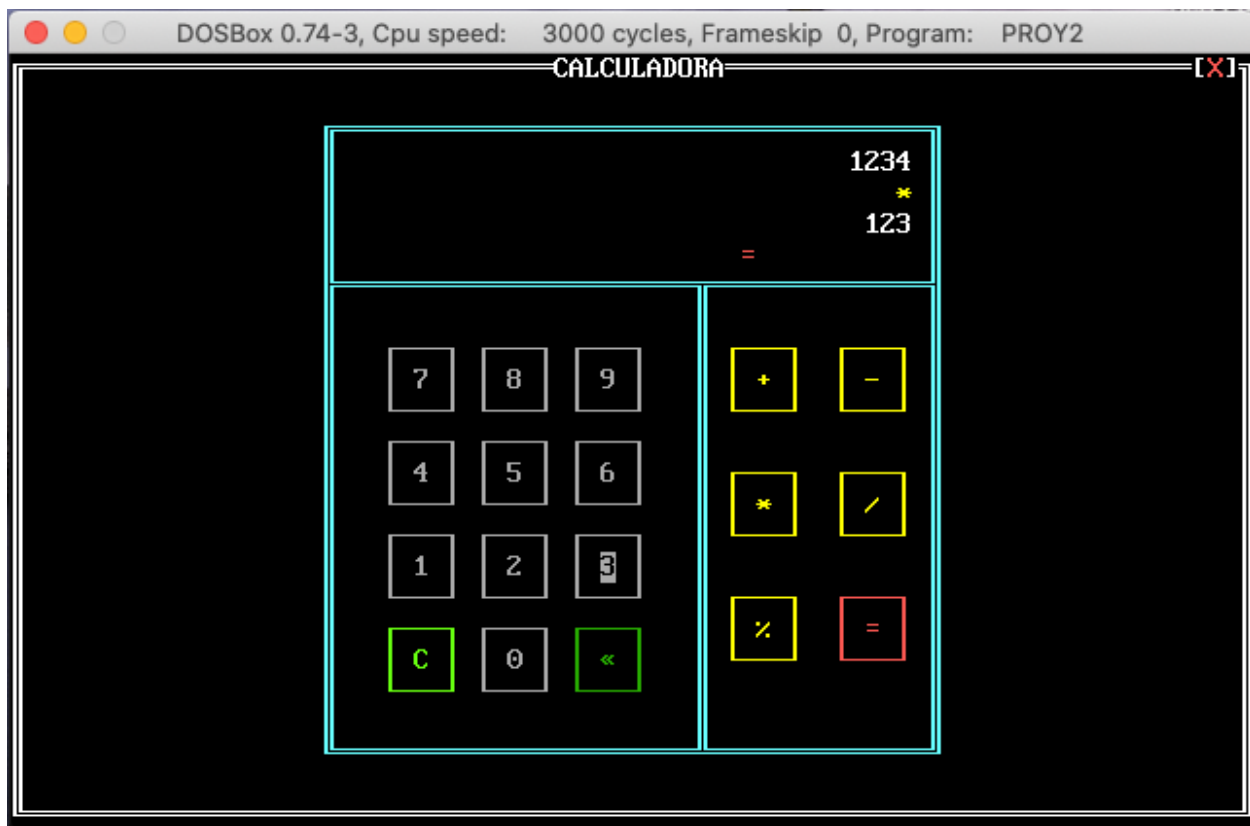
“Division entre 0”, “NaN”, etc.). Lo importante es que el programa no se detenga cuando suceda una división entre cero internamente.

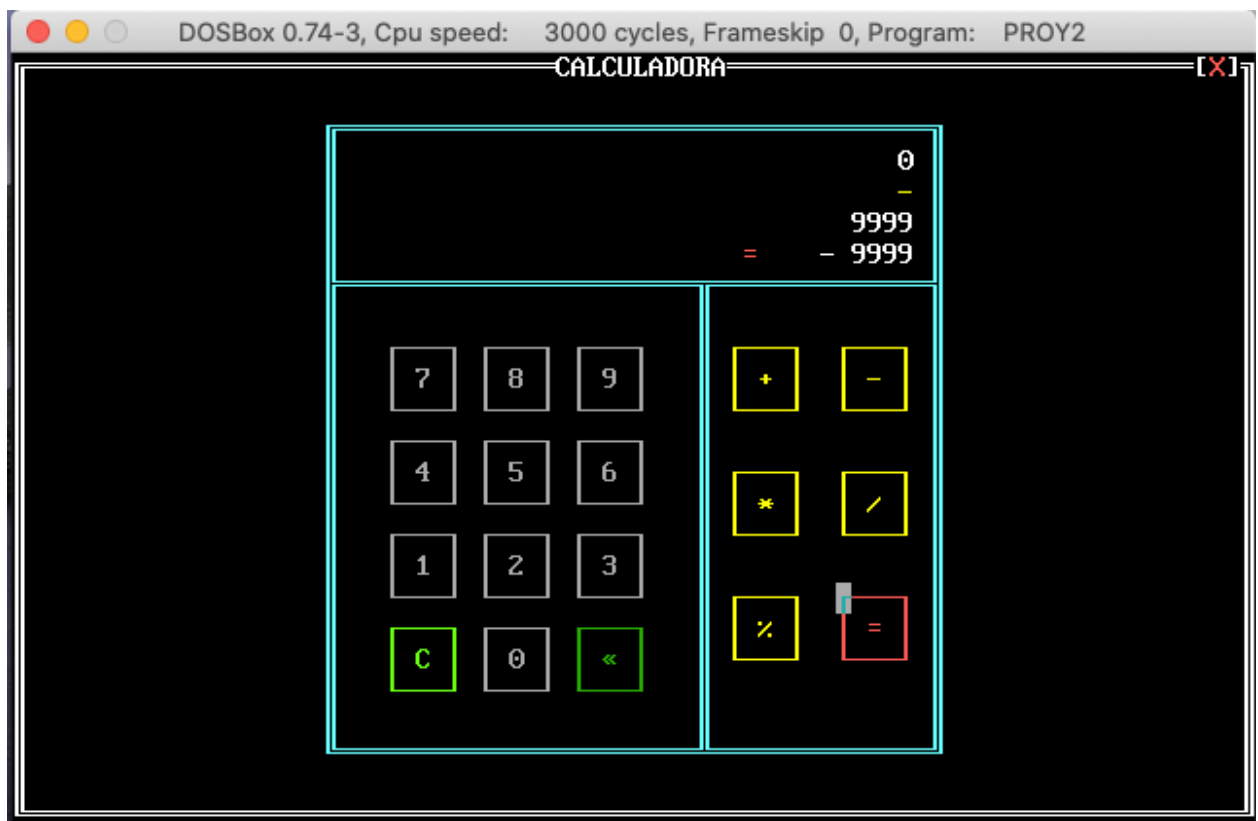
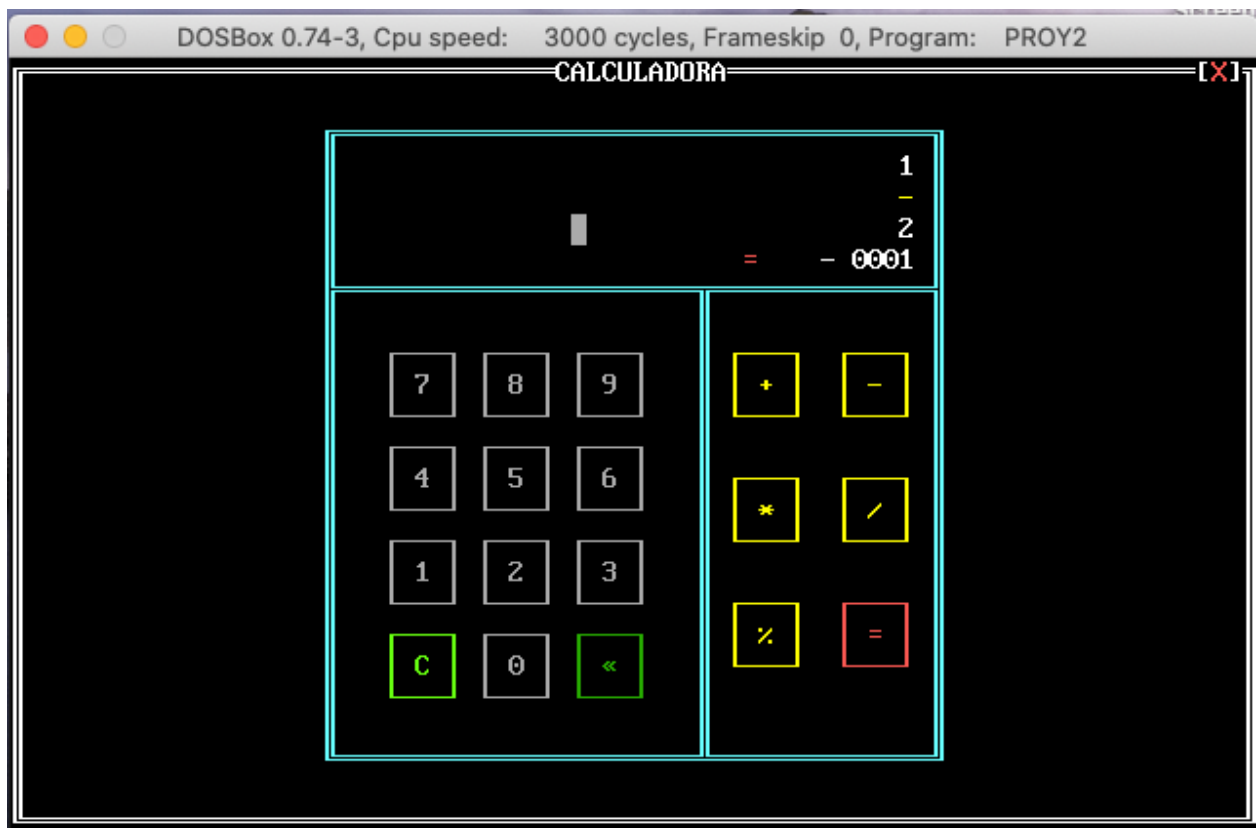
Ejemplos:

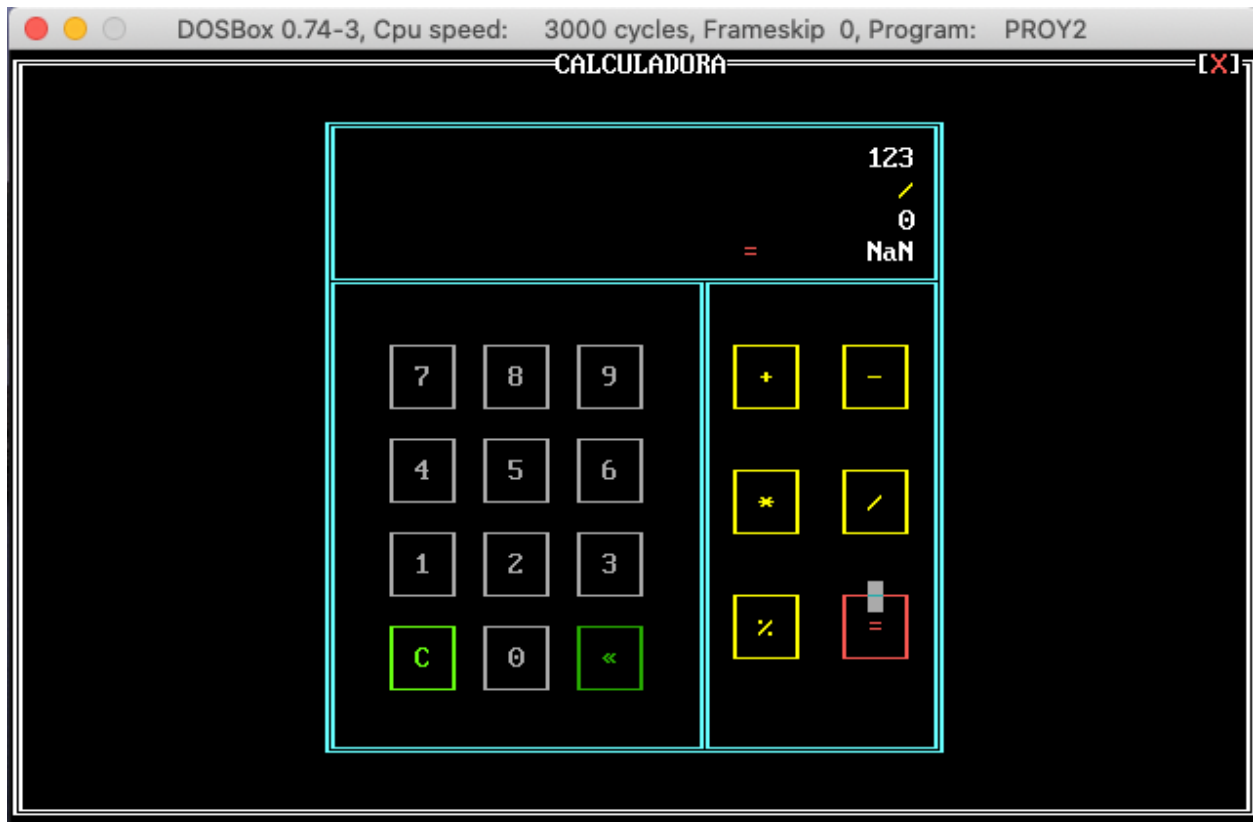












El programa final no necesariamente debe funcionar igual que el ejemplo anterior, pero sí debe realizar y mostrar lo que se solicita en este documento.

Consideraciones:

- Antes de una división se puede comparar el valor del divisor para determinar si éste es 0 o no. De esa manera se puede determinar y evitar si la división se hace entre 0.
- El guión de un número negativo no se puede imprimir automáticamente utilizando lenguaje ensamblador, éste debe imprimirse manualmente.
- Para calcular si un número entero con signo es negativo, se puede realizar una comparación entre 80h (8 bits) u 8000h (16 bits). Si el número es mayor o igual a ese valor, entonces podemos decir que es negativo. Otra forma de revisar el valor es utilizando la instrucción CMP, se puede comparar el número con 0 y se puede utilizar el salto condicional JL para determinar si el primer número con signo es menor al segundo. Por ejemplo:
var1 = FFFFh = 65535d (entero sin signo) = -1d (entero con signo)

```

cmp [var1],0    ;compara [var1] con 0
jl negativo     ;si [var1], como entero con signo, es menor a 0, entonces salta a la etiqueta
;negativo
Jmp positivo
negativo:

```



```
call IMPRIME_GUION  
neg [var1]  
positivo:  
call IMPRIME_VAR1
```

Dentro de esta misma consideración, se puede obtener el equivalente positivo del número negativo con signo calculando su complemento a 2.

- Para determinar si el clic del mouse se hizo dentro de un botón, se deben conocer las coordenadas del mouse en donde se hizo clic y revisar si se encuentra dentro de los límites. Del botón. Para ello es necesario realizar comparaciones.
- La pantalla funciona en modo texto por default, con resolución de 80x25, es decir, 80 columnas x 25 renglones. Las coordenadas del mouse se obtienen en la mayor resolución de la pantalla posible en DOS, ésta es 640x200, 640 columnas x 200 renglones. Para obtener las coordenadas del mouse en resolución 80x25, es posible haciendo una división de su valor tanto de la columna como del renglón entre 8.
- El primer renglón se indica con 0 y el último con 24. La primera columna se indica con 0 y la último con 79.
- Las coordenadas de la pantalla en modo texto inician en (0,0) en la esquina superior izquierda, y ascienden hacia abajo en renglones y hacia la derecha en columnas, de tal manera que la coordenada (79,24) es la esquina inferior derecha de la pantalla.
- Las dimensiones de la calculadora en el código base son 40 columnas por 21 renglones. La coordenada de su esquina superior izquierda es (20,2) y la coordenada de su esquina inferior derecha es (59,22)
- En el código base, cada botón tiene dimensiones de 5 columnas x 3 renglones y puede tener un carácter centrado y un color. Revisar los comentarios dentro del código base para más detalles.
- Investigar int 10h para manejo de video e int 33h para el manejo del mouse si se quieren saber más detalles del funcionamiento de cada una.

Entregables:

La entrega de proyecto se puede hacer de manera individual o en equipo de hasta 3 integrantes.

Para la entrega del proyecto, se deberá realizar una muestra 'presencial' en la que los integrantes del equipo mostrarán el funcionamiento de su proyecto y responderán a los cuestionamientos hechos por el profesor. De manera aleatoria, el profesor puede seleccionar a un integrante del equipo y hacer un cuestionamiento. Si la respuesta no es satisfactoria, puede tener repercusión en el equipo e individualmente.

Además de la muestra 'presencial', se deberá subir a la carpeta de Google Drive correspondiente lo siguiente:

- Documento
 - Introducción
 - Descripción y planteamiento del problema
 - Desarrollo
 - Planteamiento y justificación de la solución
 - Diagrama de flujo y pruebas de escritorio
 - Conclusión
 - Conclusiones individuales (por cada integrante del equipo)
- Código fuente (extensión .asm)
- Archivo ejecutable (extensión .exe)
- Archivo de texto plano (.txt) con el nombre de los integrantes del equipo.

El número de equipo se asignará de manera previa a la muestra presencial.

La fecha de la muestra presencial será el día 25 de mayo de 2020, en horario de clase*.

La fecha límite para subir el documento del proyecto es el día 24 de mayo de 2020, a las 23:59:59 hrs*.

La fecha límite para subir los archivos .asm y .exe es un instante antes de que comiencen las muestras presenciales. Se tomarán dichos archivos para ejecutar el programa durante la muestra*.

Archivos recibidos después de los horarios indicados no se tomarán en cuenta.

*Las fechas pueden cambiar. Si esto sucede, se avisará con anticipación.