**Instituto Tecnológico de Costa Rica**

**Sede San Carlos**

**Carrera de Ingeniería en Computación**

**Curso Fundamentos de organización de computadoras**

**Profesora Ing. Carlos Castillo**

**Primer proyecto programado**

**“Calculadora”**

**Estudiante**

**Marcos Rodríguez Ovares**

INDICE

[Objetivo 3](#_Toc358490125)

[Introducción 3](#_Toc358490126)

[Análisis del problema 3](#_Toc358490127)

[Descripción de la solución 3](#_Toc358490128)

[Análisis de resultados 5](#_Toc358490129)

[Capturas de pantalla 6](#_Toc358490130)

[Conclusiones y recomendaciones 10](#_Toc358490131)

[Referencias 10](#_Toc358490132)

# Objetivo

Utilizar el emulador EMU­8086 y los fundamentos del lenguaje ensamblador  para crear una pequeña aplicación básica de calculadora.

# Introducción

Una **calculadora** es un dispositivo que se utiliza para realizar cálculos aritméticos. Aunque las calculadoras modernas incorporan a menudo un ordenador de propósito general, se diseñan para realizar ciertas operaciones más que para ser flexibles. (Wikipedia La Enciclopedia Libre, 2013 )

Para este propósito una calculadora será expresada como un programa el cual realizara cálculos básicos.

# Análisis del problema

Se debe crear un programa que realice las cuatro funciones básicas de una calculadora, las cuales son: sumar, restar, dividir y multiplicar. Estas operaciones serán efectuadas entre dos operandos de un máximo de cuatro dígitos cada uno.

El programa deberá solicitar los valores de los dos operandos, seguido del símbolo que representa una de las cuatro antes mencionadas operaciones. Seguido de esto, debe desplegar el resultado correspondiente y terminar de forma correcta el programa.

# Descripción de la solución

Para una apropiado funcionamiento se hacen uso de las funciones: “SCAM\_NUM” y “PRINT\_NUM” las cuales son tomadas de ejemplos propios del emmu 8086.

También se crea el proceso “SPACE” el cual imprime un salto de línea para mantener un orden apropiado.

Para iniciar correctamente, se definen los mensajes que se van a mostrar al usuario, los cuales son los siguientes:

1. "Inserte el primer operando:"
2. "Inserte el segundo operando:"
3. "Digite el símbolo de la operación"
4. "El resultado es:"
5. "Presione una tecla para terminar"
6. "Esc -->Salir Enter --> Continuar:"
7. "NO SE PUEDE TENER UN 0 EN UNA DIVISION"

También se definen las variables que se van a necesitar, “var1” la cual almacenara el primer valor, “var2” que almacenara el segundo, “operator” guardara el símbolo correspondiente a la operación deseada y por ultimo “result” almacenara e resultado.

La etiqueta “main”, trabaja como la función principal, esta procede de la siguiente manera:

1. Imprime el primer mensaje solicitando el primer operando.
2. Se invoca a la función “SCAM\_NUM”
3. Se almacena el primero operando en la variable “var1”
4. Se invoca la función “SPACE”
5. Imprime el primer mensaje solicitando el segundo operando.
6. Se invoca a la función “SCAM\_NUM”
7. Se almacena el primero operando en la variable “var2”
8. Se invoca la función “SPACE”

Se define la etiqueta “get\_operator” la cual realizara el cálculo matemático de la siguiente manera:

1. Se invoca la función “SPACE”.
2. Imprime el primer mensaje solicitando el símbolo que representa una operación.
3. Compara el símbolo, si es un “+” realiza una suma, si es un “-” realiza una resta, si es un “|” realiza una división entera y si es un “.” Realiza una multiplicación.

Para realizar las operaciones se definieron las siguientes funciones:

**do\_add:** suma dos valores.

**do\_mul:** multiplica dos valores.

**do\_div:** divide dos valores (división entera).

**do\_sub:** resta un valor a otro.

Para finalizar se define la etiqueta “printRes” la cual se encarga de imprimir el resultado de la siguiente manera:

1. Se invoca la función “SPACE”.
2. Se imprime el mensaje de resultado.
3. Se invoca la función “PRINT\_NUM”.
4. Se invoca la función “SPACE” dos veces.
5. Se imprime un mensaje de presionar una tecla para salir.
6. Espera una tecla de consola y termina la ejecución del programa.

# Análisis de resultados

Con base en las indicaciones:

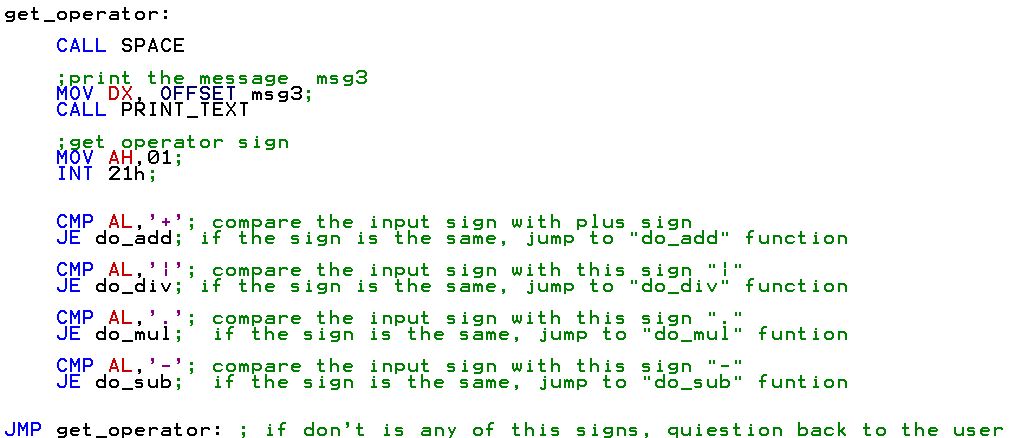
Se solicitan y guardan correctamente los operandos, se guarda y compara el símbolo insertado con su respectiva operación, se realiza correctamente la operación y por último se muestra el resultado obtenido.

Al solicitar un operando solo se pueden insertar números de cuatro dígitos como máximo, de lo contrario se vuelve a solicitar, si el usuario digita “ENTER” sin insertar un número, el programa lo reconoce como un cero, lo cual no causaría problema alguno, con excepción claro está de la división por cero.

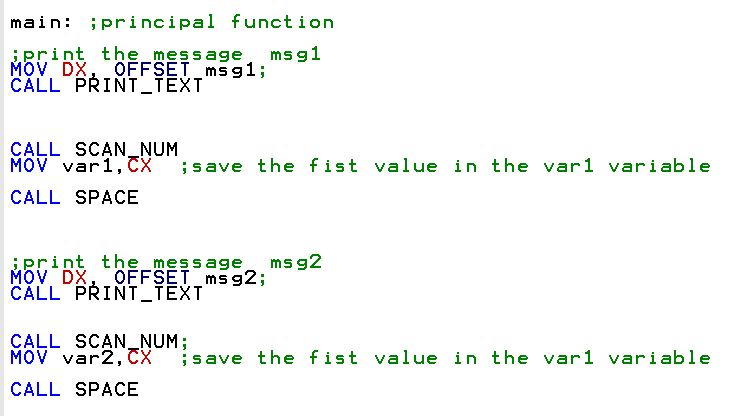
Cuando se solicita una operación, el programa solo acepta los cuatro definidos para las operaciones, cualquier otro símbolo o numero recurrirá en una nueva solicitud.

# Capturas de pantalla

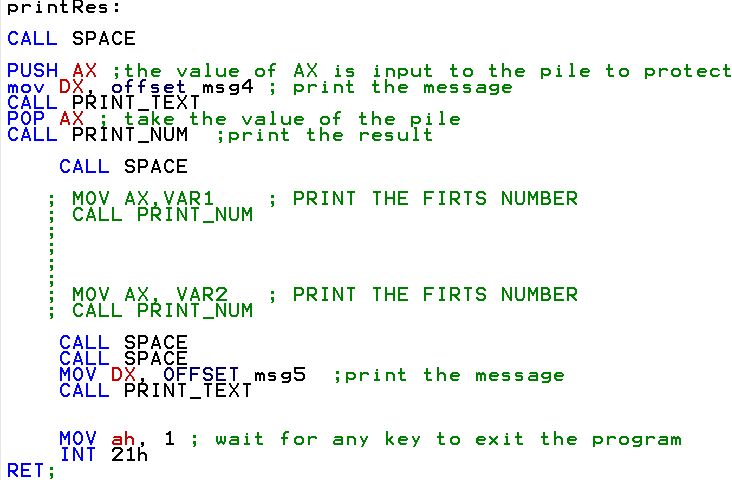
Etiqueta “get\_operator”



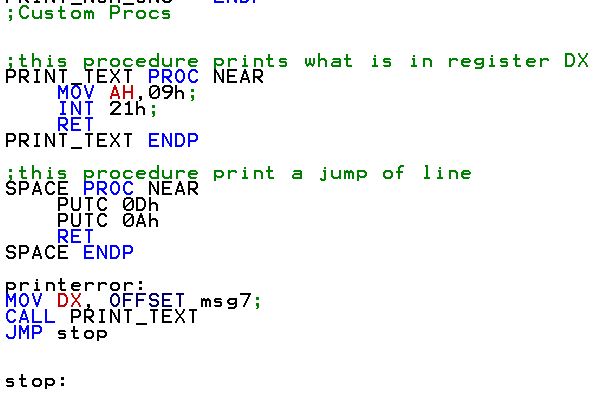
Etiqueta principal “main”



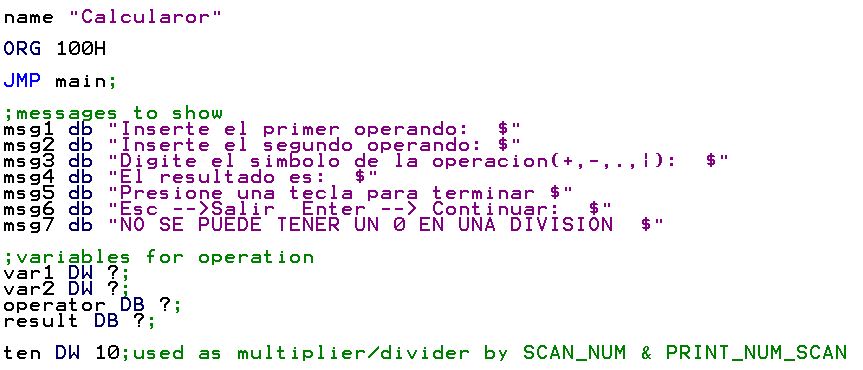
Etiqueta “printRes”



Procesos “Print\_Text”, “Space”, “printerror”



Variables y mensajes



Conclusiones y recome**ndaciones**

Un objetivo que no se logro fue el de evitar el error al dividir por cero, aunque esto no era parte fundamental del proyecto, se considera básico en una calculadora.

A pesar de esto los resultados cumplen con el objetivo del proyecto y la mayoría de los lineamientos.

Cabe resaltar que pese a que la dificultad es mínima, el lenguaje ensamblador requiere un amplio conocimiento en distintas formas no convencionales de realizar una labor, el análisis de los procesos es más profundo y adicionalmente se conoce a fondo las propiedades de un computador.

# Referencias

*Wikipedia La Enciclopedia Libre*. (13 , de mayo de 2013 ). Obtenido de http://es.wikipedia.org/wiki/Calculadora