

Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Laboratorio Arquitectura de Computadoras y  
Ensambladores 1  
Aux. Herberth Argueta      Sección: A  
Stefany Samantha Abigail Coromac Huevo  
201801182



## MANUAL TECNICO

### objetivo general:

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

### objetivos Específicos:

- Aplicar el conocimiento de operaciones básicas a nivel ensamblador.
- Conocer el funcionamiento de las interrupciones.
- Comprender el uso de la memoria en los programas informáticos.
- Consolidar los conocimientos de escritura\lectura de archivos.

### Descripción practica 4 (ensamblador)

La practica consiste en una aplicación de consola en lenguaje ensamblador, la cual tendrá la funcionalidad de una calculadora en cual podrá hacer operaciones básicas, suma, resta, multiplicación y división.

### MACROS UTILIZADAS PARA LA PRACTICA

- **OpenF:** Abre un archivo.
- **CloseF:** Cierra un archivo
- **WriteF:** Escribe sobre un archivo el numero de bytes que se desee.
- **CrearF:** Crea un nuevo archivo.
- **ComprarCadenas:** Compara si dos cadenas son iguales
- **calcularMedia:** Calcula la media de los resultados obtenidos.
- **calcularMayor:** Calcula el mayor de los resultados obtenidos
- **getChar:** Lee un char desde el teclado.
- **buscandoID:** busca un ID para mostrar el resultado
- **ConvertirAscci:** Convierte un buffer a su equivalente en ASCII.

- **ConvertirString:** Convierte el numero guardado en el registro ax para poder mostrarlo en pantalla.
- **Transferir:** Transfiere lo que hay en un buffer a otro.
- **crearReporte:** Crear el reporte con sus nuevos datos.
- **showDate:** Obtiene la fecha para colocarlo en el reporte.
- **ObtenerOperaciones:** Obtiene las operaciones para colocarlo en el reporte
- **ObtenerResultados:** Obtiene los resultados para colocarlo en el reporte
- **limpiarBuffer:** Limpia un arreglo colocando '\$' en cada posición
- **leyendoJSON:** Es el analizador que obtiene los resultados de las operaciones.
- **print:** Imprime en pantalla
- **opcion1:** Es la opción para cargar un archivo.

## variables y cadenas utilizadas para la practica

```
.model small
.stack 100h
.data
; ----- SEGMENTO DE DATOS -----
encabezado db 0ah, 0dh, '===== UNIVERSIDAD DE SAN CARLOS DE GUATEMALA',
0ah, 0dh, 'Facultad de Ingenieria',
0ah, 0dh, 'Ciencias y Sistemas',
0ah, 0dh, 'Arquitectura de Computadores y Ensambladores 1',
0ah, 0dh, 'Nombre: Stefany Samantha Abigail Coromac Huezo',
0ah, 0dh, 'Carne: 201801182',
0ah, 0dh, 'Seccion: A',
0ah, '$'
menu db 0ah, 0dh, '===== ',
0ah, 0dh, '1. CARGAR ARCHIVO',
0ah, 0dh, '2. CONSOLA',
0ah, 0dh, '3. SALIR',
0ah, 0dh, '===== ',
0ah, 0dh, 'Ingrese Opcion:', '$'

encCargarArchivo db 0ah, 0dh, '===== CARGAR ARCHIVO ====='
encConsola db 0ah, 0dh, '===== CONSOLA ====='
inicioConsole db 0ah, 0dh, '>>', '$'
espacio db 20h, '$'

saltoLinea db 0ah, 0dh, '$'
finObjeto db 0
comandoConsola db 30 dup('$')
comandoConsola2 db 30 dup('$')
showMedia db 'media', '$'
showMediana db 'mediana', '$'
showModa db 'moda', '$'
showMenor db 'menor', '$'
showMayor db 'mayor', '$'
showMayor2 db 'mayor', '$'
media dw 0
mediana dw 0
moda dw 0
menor dw 0
mayor dw 0

bufferMediaR dw 10 dup('$')
bufferMedianaR dw 10 dup('$')
bufferModaR dw 10 dup('$')
bufferMenorR dw 10 dup('$')
bufferMayorR dw 10 dup('$')
```

```

;-----LECTURA DEL ARCHIVO .JSON -----
msgErrorOpen db 0ah, 0dh, 'Error al Abrir el archivo', '$'
msgErrorRead db 0ah, 0dh, 'Error al leer el archivo', '$'
msgErrorCreate db 0ah, 0dh, 'Error al crear el archivo', '$'
msgErrorWrite db 0ah, 0dh, 'Error al escribir archivo', '$'
msgArchivoLeido db 0ah, 0ah, 0dh, '> Archivo Leido con exito! ', '$'
msgFinOperacion db 0ah, 0dh, 'FIN DE OPERACION ', '$'
msgCargarArchivo db 0ah, 0dh, ' INGRESE RUTA: ', '$'
rutaArchivo db 100 dup(?)
bufferLectura db 10000 dup('$')
limpiarD db 21 dup('$')
bufferEscritura db 20 dup(' ')
handleFichero dw ?
dividirpor db 0

suma1 db 0ah, 0dh, 'SUMA', '$'
resta1 db 0ah, 0dh, 'RESTA', '$'
multiplicacion1 db 0ah, 0dh, 'MULTIPLICACION', '$'
division1 db 0ah, 0dh, 'DIVISION', '$'

;=====Variables para Lectura de JSON (ANALIZADOR) =====

estado db 0
contadorPadre db 0
contadorNumero db 0
inicioArchivo db 0
contadorLlaves db 0
bufferAux db 30 dup('$')
finOpe db 0
msgRegresarPila db 0ah, 0dh, 'Regresando registros a la Pila', '$'
msgRevisarPila db 0ah, 0dh, 'Revisando pila ', '$'
msgIDoperacion db 0ah, 0dh, 'Este es el ID de una nueva operacion: ', '$'
msgResultado db 0ah, 0dh, 'Resultado ', '$'
msgnumeroNegativo db 0ah, 0dh, 'Es un numero negativo', '$'
negativo db 0
auxiliar dw 0

;=====OPERACIONES=====
resultados db 30 dup(0)
totalOperaciones dw 0
operaciones dw 20 dup('$')
total dw 0
imprimirNumero db 30 dup('$')
numero1 db 100 dup('$')
numero2 db 100 dup('$')
nombrePadre db 20 dup('$'), '$' ; AQUI LO TENGO COMO LO GUARDO DESDE EL ARCHIVO
extension db '.json', '$'
nombreReporte db 20 dup('$'), '$' ; NOMBRE DEL REPORTE CON LA EXTENSION
nombreOperacion db 12 dup('$')

```

```

; =====VARIABLES PARA REPORTE =====
comillas db '''
finReporte db 0ah, '}',
           0ah, '}'
inicioReporte db '{',
              0ah, ' "reporte":',
              0ah, '{',
              0ah, " "alumno":",
              0ah, '{',
              0ah, " "Nombre":"Stefany Samantha Abigail Coromac Huezo",',
              0ah, " "Carne":"201801182",',
              0ah, " "Seccion":"A",',
              0ah, " "Curso":"Arquitectura de Computadores y Ensambladores 1",',
              0ah, '}',
              0ah, " "fecha":",
              0ah, '{',
              0ah, " "Dia":',
              0ah, " "Mes":',
              0ah, " "Año":',
              0ah, '}'
bufferMes db 0ah,
bufferAnio db 0ah,
cerrarObject1 db 0ah,
inicioHora db 0ah,
           0ah, ' "hora":',
           0ah, '{',
           0ah, " "Hora":',
           0ah, " "Minutos":',
           0ah, " "Segundos":',
bufferMinutos db 0ah,
bufferSegundos db 0ah,
inicioResultados db 0ah, ' "resultados":',
           0ah, '{',
           0ah, " "Media": ',
bufferMediana db 0ah, " "Mediana": "Aquí estuviera si tan solo lo hubiera hecho"'
bufferModa db 0ah, " "Moda": "Aquí estuviera si tan solo lo hubiera hecho"'
bufferMenor db 0ah, " "Menor": "Aquí estuviera si tan solo lo hubiera hecho"'
bufferMayor db 0ah, " "Mayor": "Aquí estuviera si tan solo lo hubiera hecho"'
inicioPadre db 0ah, ' "operaciones":'
comilla db 0ah, '""'
cierreComillas db ':'
cierreComillas2 db ':', '$'
inicioArreglo db 0ah, '['
finArreglo db 0ah, ']'
coma db ','
coma2 db ',','$'
inicioOR db 0ah, 09h, 09h, 09h, '{',
           0ah, 09h, 09h, 09h, 09h, '','', '$'
finOR db 0ah, 09h, 09h, 09h, '}', '$'

```