

## Práctica 4

---

### Objetivos

#### 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 escritos en ensamblador
- Aplicar el manejo de archivos a bajo nivel
- Aplicar el uso de registros bandera
- Manejar entrada, proceso y salida de datos numéricos en ensamblador

---

### Descripción

Se requiere el desarrollo de una aplicación en consola, utilizando programación a bajo nivel (lenguaje ensamblador), que tenga las funcionalidades de una calculadora que realice las siguientes operaciones:

- Suma
- Resta
- Multiplicación
- División
- Potenciación

Además de poder obtener los datos estadísticos de los números ingresados, por medio de archivo o directamente; estos serán:

- Media
  - Mediana
  - Moda
  - Cantidad de números pares
  - Cantidad de números impares
  - Cantidad de números primos
-

## Especificaciones

### Símbolos a utilizar

Operación	Consola	Archivo
Suma	+	add
Resta	-	sub
Multiplicación	x	mul
División	/	div
Potenciación	^	pow

### Estadísticos

Estadístico	Observaciones
Media	Salida parte entera
Mediana	Salida parte entera
Moda	
Números pares	
Números impares	
Números primos	Se debe calcular

Los estadísticos se calculan sobre las entradas recibidas y no sobre los resultados, finales o parciales, de las operaciones.

---

## Desarrollo de la práctica

### Identificación del desarrollador

Cada vez que se ejecute la práctica se deberá mostrar la identificación del desarrollador, ejemplo:

```
Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Arquitectura de Compiladores y ensambladores 1
Sección <A|B>
< Nombre del desarrollador >
< Registro Académico >
```

Luego de la identificación deberá esperar a que se presione Enter para poder continuar hacia el menú.

---

## Menú

El menú del desarrollo debe incluir las siguientes opciones

1. Calculadora
2. Archivo
3. Salir

### Calculadora

Mostrará en consola la opción de realizar operaciones matemáticas.

### Archivo

Permitirá introducir la ruta del archivo de entrada en el que se almacenaron las operaciones a realizar.

### Salir

De no haber generado el reporte de ejecución (explicado más adelante) se generará en este momento y terminará la ejecución del programa.

---

## Detalle Funciones del menú

### Calculadora

Permitirá realizar operaciones matemáticas, desde línea de comandos, en notación polaca inversa e ingresar instrucciones que muestran reportes específicos.

Ejemplos de operaciones directas de calculadora

Operación:  $8 + 25$

Calculadora

```
> 8  
> 25  
> +
```

donde se ingresó primero el 8, luego el 25, por último el signo "+"; al presionar enter borrará el contenido anterior y mostrará únicamente el resultado que queda disponible en caso de ingresar algún otro número y signo que lo vaya a utilizar

Calculadora

```
> 33  
>
```

Operación:  $((12 + (8 ^ 2)) / 4)$

```
Calculadora
> 12
> 8
> 2
> ^
```

donde se ingresó primero el 12, luego el 8 seguido del 2 y el signo de potencia, esto operará el  $8 ^ 2$  primero; mostrando en pantalla el resultado de la última operación y el primer número ingresado

```
Calculadora
> 12
> 64
> 
```

quedará en espera de nuevos datos u operar los dos en pila, para este caso ingresaremos el signo "+" y lo aceptamos con enter

```
Calculadora
> 12
> 64
> +
```

```
Calculadora
> 76
> 
```

Por último operamos la división entre 4

```
Calculadora
> 76
> 4
> /
```

```
Calculadora
> 19
> 
```

## Instrucciones reconocidas

Instrucción	Función
AC	<ul style="list-style-type: none"><li>- Reiniciará todos los estadísticos tomados hasta el momento</li><li>- Iniciará con una nueva operación</li><li>- Si se ha cargado una operación desde archivo volverá a la posibilidad de usar la consola como calculadora o evaluar un nuevo ID</li></ul>

Instrucción	Función
PRINT MEDIA	Mostrará la media de los números reconocidos al momento
PRINT MODA	Mostrará la moda de los números reconocidos al momento
PRINT MEDIANA	Mostrará la mediana de los números reconocidos al momento
PRINT IMPARES	Mostrará la cantidad de números impares reconocidos
PRINT PARES	Mostrará la cantidad de números pares reconocidos
PRINT PRIMOS	Mostrará la cantidad de números primos reconocidos
EVAL [ID]	<ul style="list-style-type: none"> <li>- [ID] será reemplazado por el nombre de la operación cargada desde archivo</li> <li>- Se analizará la operación en cuestión y realizará todos los cálculos</li> <li>- Mostrará el resultado de la evaluación completa de la operación</li> <li>- quedará en espera de la instrucción PRINT deseada</li> <li>- no podrá realizar cálculos matemáticos si se ha cargado un ID</li> </ul>
END	Salir al menú principal

Ejemplo de instrucciones:

```

Calculadora
> EVAL oper1
El resultado es: 33

Calculadora
> PRINT media
La media desde archivo es: 7

Calculadora
> 

```

Si se están haciendo cálculos y se usa alguna instrucción PRINT solo se podrán seguir haciendo cálculos si se ingresa AC para volver a empezar de cero con una nueva operación.

Si se trata de una operación desde archivo se refiere únicamente a los datos de la operación referenciada por nombre, aunque el archivo contenga varias operaciones.

Si existiese más de una moda se tomará la primera.

Se sugiere el uso de una pila para el manejo de los operandos.

## Archivo

Permite ingresar el nombre del archivo que contiene las operaciones a evaluar

Archivo

Nombre de archivo:

El contenido del archivo será JSON agrupando los operandos en pares y tendrá operaciones en notación polaca inversa; el padre siempre será "CALCULOS" y deberá validarse para poder operar el archivo.

Ejemplo del archivo de entrada

```
{
  "calculos": [
    "oper1": {
      "var1": 8,
      "var2": 25,
      "oper": "add"
    },
    "ejemplo2": {
      "var1": {
        "var1": 12,
        "var2": {
          "var1": 8,
          "var2": 2,
          "oper": "pow"
        },
        "oper": "add"
      },
      "var2": 4,
      "oper": "div"
    },
    "calculos": {
      "var1": {
        "var1": {
          "var1": 5,
          "var2": 9,
          "oper": "add"
        },
        "var2": 2,
        "oper": "mul"
      },
      "var2": {
        "var1": 6,
        "var2": 5,
        "oper": "mul"
      },
      "oper": "add"
    }
  ]
}
```

## Reporte de ejecución

Se generará un archivo con contenido de tipo JSON que detalle los siguientes aspectos:

- Datos del alumno
  - Nombre
  - Carnet
  - Curso
  - Sección
- Fecha de generación de reporte
  - Día
  - Mes
  - Año
- Hora de generación de reporte
  - Hora
  - Minuto
  - Segundo
- Resultados:
  - media
  - mediana
  - moda
  - pares
  - impares
  - primos
- Operaciones:
  - [Arreglo de operaciones evaluadas]

Ejemplo:

```
{
  "reporte": [
    "Datos": {
      "nombre": "...",
      "carnet": "20XYABCDE",
      "curso": "Arquitectura de compiladores y ensambladores 1",
      "sección": "A|B"
    },
    "Fecha": {
      "Día": 09,
      "Mes": 03,
      "Año": 2022
    },
    "Hora": {
      "hora": 03,
      "minuto": 05,
      "segundo": 59
    }
  ],
}
```

```

    "Estadísticos":{
        "media":7,
        "mediana":6,
        "moda":8,
        "impares":4,
        "pares":7,
        "primos":3
    },
    "Operaciones":[
        {
            "oper1": 33
        },
        {
            "ejemplo2": 19
        },
        {
            "calculos": 58
        }
    ]
}

```

En el caso que no se haya cargado un archivo de entrada, en operaciones habrá únicamente una y su resultado será el último generado por la calculadora.

Siempre se mostrarán los resultados del último análisis generado, ya sea de archivo o de cálculos directos.

Los cálculos directos también generan los datos estadísticos.

## Referencias

Titulo	Referencia
Tabla ASCII	<a href="https://elcodigoascii.com.ar">https://elcodigoascii.com.ar</a>
Libro de Texto	Brey, Barry B. Microprocesadores Intel 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro Processor, Pentium II, Pentium III, and Pentium 4 Arquitectura, Programación e Interfaz. Pearson Prentice Hall. Séptima Edición.
Libro sugerido	Assembly language for x86 processors. Irvine - Pearson - 2015

## Requerimientos mínimos

- Lectura de archivo (para el modo archivo)
- Generación de reporte
- Se debe entregar los manuales técnico y de usuario, de lo contrario se asumirá que el estudiante copió.



Debido a la naturaleza de ensamblador, los resultados completos solo se pueden evaluar si se logran cumplir con los dos primeros puntos

## Observaciones y Restricciones

- **La realización de la práctica es de forma individual**
- Se evaluarán números de 16 bits con signo
- El código del programa debe ser estrictamente ensamblador, no se permite el uso de alguna librería.
- Se debe presentar el proyecto en Dosbox.
- Lenguaje ensamblador a utilizar MASM 6.11
- No está permitido el uso de estructuras de control if o if else, while, repeat, for.
- No está permitido el uso de STRUCT
- El día de la calificación se harán preguntas, modificación de código sobre aspectos utilizados en la elaboración del proyecto, las cuales se considerarán en la nota final.
- Subir todos los archivos para que la práctica funcione así como los manuales para antes de 23:59 horas del jueves 31 de marzo.
- **Copias parciales o totales tendrán una nota de 0 puntos y los involucrados serán reportados a la Escuela de Ciencias y Sistemas**
- Todos estos entregables deben de ser adjuntados en un archivo de extensión rar. El archivo debe ser nombrado como [ACE1]P4\_#Carnet.rar.