
	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

GUÍA DE LABORATORIO DE ESTRUCTURAS DE DATOS						
Unidad Didáctica: Práctica No.1 Introducción a las estructuras de datos						
Eje Temático: Introducción a las estructuras de datos						
No. Guía		1		<b>Resultados de Aprendizaje de la Unidad Didáctica:</b>  Utiliza correctamente las variables y sentencias de control -if, for, while- para la implementación de un programa Relacionar las competencias descritas en el plan de curso.		
1 sesiones		1 Semana				
Horas de Trabajo						
Trabajo con Docente		Trabajo Autónomo				
3		6				
Tipo de trabajo						
Grupal	X	Ind		Laboratorio Requerido	Asistido por computador / Laboratorio de Informática	
Introducción						
<i>Estos ejercicios se deben solucionar en lenguaje Java utilizando la herramienta Netbeans.</i> Enfocar al estuudio de Ingeniería de Software en el aprendizaje acerca de las Estructuras de Datos y de la manipulación la información a través de ordenamiento, búsqueda, organización y el conocimiento de los métodos de acceso, porque una estructura de datos eficiente siempre se relaciona con el análisis del rendimiento de algoritmos de programación.						
<b>Subtemas:</b> Introducción a las estructuras de datos: <ul style="list-style-type: none"><li>Definición, uso y aplicación de las estructuras de datos en ingeniería.</li><li>Conceptos básicos de programación.</li></ul>						
<b>Sesión 1. ¿Qué haremos en la sesión? ¿Cómo lo haremos?</b>						
1. Revisar y comprender los requerimientos de la guía de laboratorio. Pedir al profesor las aclaraciones del caso: <ul style="list-style-type: none"><li>Leer marco conceptual o referencial.</li><li>Análisis de los requerimientos del informe de laboratorio.</li></ul>						
2. Abrir el entorno de trabajo. Hacer uso de las herramientas (software) sugeridas por el profesor.						
3. Elaborar el informe de laboratorio: <ul style="list-style-type: none"><li>Diligenciar los datos básicos del informe</li><li>Estudiar los presaberes requeridos</li></ul>						

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

- c. Realizar la actividad de trabajo independiente
- d. Diligenciar los resultados obtenidos
- e. Diligenciar el Análisis e Interpretación de resultados
- f. Escribir las Conclusiones

4. Desarrollar los siguientes puntos:

a) Mencione 10 situaciones en la que se aplican las estructuras de datos.

b) Haciendo uso de estructuras de datos estáticos (vectores y matrices) sin POO, Generar un Algoritmo que guarda una matriz de N vehículos (Mazda, Toyota, etc.) con sus respectivos precios durante los últimos 5 años (2019, 2020, 2021, 2022, 2023), los cuales son ingresados por teclado (validar que el valor que se introduce sea positivo, si introduce valores negativos o cero debe mostrar mensaje que solo se permiten valores positivos y permitirle digitar nuevamente el precio). Mostrar:


- Mediante una función/método mostrar la Lista de autos con sus respectivos precios de cada año
- Mediante una función/método mostrar los autos más baratos de cada año
- Mediante una función/método mostrar el promedio de los autos que cuestan entre 30 y 50 millones del año seleccionado por el usuario.

c) Generar sin POO, un programa que por cada día de la semana (lunes a domingo) acumula la cantidad de ventas totales diarias de una empresa, la empresa tiene 6 sedes. Mediante funciones llenar la matriz con las ventas respectivas, haciendo uso de funciones por cada sede mostrar los días que tienen ventas totales diarias por encima de la media de la semana respectiva y el promedio de ventas total de la semana de la empresa y por cada sede. De igual manera generar un procedimiento/método que permita aumentar en un 15% las ventas diarias que estén por debajo de la media respectiva de cada sede.

4. Validar el desarrollo de cada uno de los puntos anteriores. Realizar las pruebas pertinentes para verificar que se cumple con los requerimientos.

- a) Especificar las causas de error que tuvo durante el desarrollo de la guía.
- b) Escribir las conclusiones de la práctica realizada.
- c) Incluir la bibliografía consultada. Normas APA.
- d) Incluir los anexos
- e) Validar que el informe de laboratorio esté completamente diligenciado.

5. Entregar informe de laboratorio para revisión.

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

¿Cuales fueron los aprendizajes obtenidos al realizar esta guía?, liste como mínimo 3 aprendizajes y relaciónelos con su futuro que hacer profesional.

¿Donde presento mayor dificultad resolviendo la guía? y como lo resolvieron cuales fueron las estrategias de solución?


#### Presaberes Requeridos

Se asume que tiene conocimientos sólidos sobre:

- Manejo de un lenguaje de programación.
- Definición de variables
- Condicionales
- Ciclos de repetición
- Definición de arreglos
- Llenado de un arreglo, recorrido e impresión de los datos que lo conforman
- Funciones y procedimientos
- Programación orientada a objetos
  - Crear una clase con sus métodos, acceder a sus atributos, modificar sus atributos
  - Definir una clase abstracta
  - TAD (Tipo Abstracto de Datos)

Adicional responda las siguientes preguntas Orientadoras

- Un arreglo unidimensional es un tipo de datos estructurado que está formado de una colección finita y ordenada de datos del mismo tipo. Es la estructura natural para modelar listas de elementos iguales. Un ejemplo práctico de seria:
  - a. Almacenar los nombres de los meses
  - b. Contar los números del 1 al 100
  - c. Calcular el número PI
  - d. Almacenar el número de los días de la semana
- Un arreglo tiene la característica de que puede almacenar a N elementos del mismo tipo y además permite el acceso a cada uno de estos elementos. Se distinguen dos partes en los arreglos:
  - e. Los elementos del vector y el índice.
  - f. Los datos del vector y el índice
  - g. El contador del vector y sus datos
  - h. Índice inicial y final .
- Para acceder a cada uno de los elementos del vector utilizamos un índice. el primer elemento se accede con el índice:

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

- a. 3
- b. 0**
- c. n
- d. n+1

### Marco conceptual o referencial \*

#### MARCO TEORICO.

**“Estructura de Datos:** La representación de la información es fundamental en ciencias de la computación y en informática. El propósito principal de la mayoría de los programas de computadoras es almacenar y recuperar información, además de realizar cálculos. De modo práctico, los requisitos de almacenamiento y tiempo de ejecución exigen que tales programas deban organizar su información de un modo que soporte procesamiento eficiente. Por estas razones, el estudio de estructuras de datos y de los algoritmos que las manipulan constituye el núcleo central de la informática y de la computación”.

*(Joyanes Aguilar, L., & Zohonero Martínez, I. (2008). Estructuras de datos en Java (1era ed.). Madrid, ES: McGraw-Hill.)*

“Una **estructura de datos** es una colección de datos que se caracterizan por su organización y las operaciones que se definen en ella. Los datos de tipo estándar pueden ser organizados en diferentes estructuras de datos: estáticas y dinámicas.

Estructuras de datos estáticas:	Estructuras de datos dinámicas:
<p>Son aquellas en las que el espacio ocupado en memoria se define en tiempo de compilación y no puede ser modificado durante la ejecución del programa:</p> <ul style="list-style-type: none"> <li>• Arrays</li> <li>• Registros</li> <li>• Cadenas</li> <li>• Conjuntos</li> <li>• Archivos</li> </ul>	<p>Son aquellas en las cuales el espacio asignado en memoria puede ser modificado en tiempo de ejecución:</p> <ul style="list-style-type: none"> <li>• Listas</li> <li>• Árboles</li> <li>• Grafos</li> </ul>


Las operaciones básicas que se pueden realizar con estructuras de datos son: adicionar, borrar, modificar, buscar, ordenar, aparear, recorrer, entre otras.

Cada estructura de datos ofrece ventajas y desventajas en relación a la simplicidad y eficiencia para la realización de cada operación. De esta forma, la elección de la estructura de datos apropiada para cada problema depende de factores como la frecuencia y el orden en que se realiza cada operación sobre los datos.

**Dato:** es la expresión general que describe los objetos con los cuales opera un algoritmo. El tipo de un dato determina su forma de almacenamiento en memoria y las operaciones que van a poder ser efectuadas con él.

**Tipo de dato:** es la propiedad de un valor (dato) que determina su dominio (qué valores puede tomar), qué operaciones se le pueden aplicar y cómo es representado internamente por el computador. Todos los valores que aparecen en un programa (algoritmo) tienen un tipo. En principio hay que tener en cuenta que, prácticamente en cualquier lenguaje de programación y por tanto en cualquier algoritmo, se podrán usar datos de los siguientes tipos de datos genéricos:

- Entero

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

- Real
- Lógico (verdadero, falso)
- Carácter
- Cadena (conjunto de caracteres)
- Definidos por el usuario según la necesidad.

**Constante:** Son datos cuyo valor no cambia durante todo el desarrollo del algoritmo.

**Variable:** Es un objeto cuyo valor puede cambiar durante el desarrollo del algoritmo. Se identifica por su nombre y por su tipo, que podrá ser cualquiera, y es el que determina el conjunto de valores que podrá tomar la variable. Ejemplos de variables en el lenguaje de programación Java.

Tipo de Dato	Nombre Variable	Descripción o comentario
int	codigo;	//variable de tipo entero
long	identificacion;	//variable de tipo entero
String	nombre;	//variable de tipo cadena
String	email;	//variable de tipo cadena
char	genero;	//variable de tipo carácter
double	salario;	//variable de tipo real
short	edad;	//variable de tipo entero
float	peso, estatura;	//variables de tipo real
boolean	decisión;	//variable de tipo booleano (true, false)
Date	fecha_nacimiento	//variable de tipo fecha


**Expresión:** Es una combinación de operadores y operandos. Los operandos podrán ser constantes, variables u otras expresiones y los operadores de cadena, aritméticos, relacionales o lógicos. Las expresiones se clasifican, según el resultado que producen, en: numéricas, alfanuméricas y booleanas.

**Control del flujo de un programa:** El flujo (orden en que se ejecutan las sentencias de un programa) es secuencial si no se especifica otra cosa. Este tipo de flujo significa que las sentencias se ejecutan en secuencia, una después de otra, en el orden en que se sitúan dentro del programa. Para cambiar esta situación se utilizan las estructuras de selección, repetición y salto que permiten modificar el flujo secuencial del programa.: Pueden ser de tipo: secuencial, selectivas (if, switch), repetitivas (mientras, hacer mientras, desde..hasta), anidadas y de salto.

- Estructura secuencial: es aquella en la cual una acción se ejecuta detrás de otra. El flujo del programa coincide con el orden físico en el que se sitúan las instrucciones.
- Estructura selectiva: Es aquella en que se ejecutan unas acciones u otras según se cumpla o no una determinada condición (if..... switch.....). La selección puede ser simple (if), doble (if) o múltiple (switch).
- Estructura repetitiva: Las acciones del cuerpo del bucle se repiten mientras o hasta que se cumpla una determinada condición (while..., do...while, for...). Es frecuente el uso de contadores o banderas para controlar un bucle."


(Joyanes Aguilar, L., & Rodríguez Baena, L., & Fernández Azuela, M. (2003). *Fundamentos de programación: libro de problemas. Algoritmos, estructuras de datos y objetos* (2a. ed.). Madrid, ES: McGraw-Hill España, 2003.)

La siguiente tabla es un completo resumen sobre los tipos de datos que se utilizan en el lenguaje de programación.

	Nombre del Proceso:	<b>CODIGO: LA-FM-001</b>
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	<b>VERSION: 7</b>
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	<b>FECHA: 15/junio/2022</b>

TIPOS DE DATOS EN JAVA	TIPOS PRIMITIVOS (sin métodos; no son objetos; no necesitan una invocación para ser creados)	NOMBRE	TIPO	OCUPA	RANGO APROXIMADO
		byte	Entero	1 byte	-128 a 127
		short	Entero	2 bytes	-32768 a 32767
		int	Entero	4 bytes	2*10 <sup>9</sup>
		long	Entero	8 bytes	Muy grande
		float	Decimal simple	4 bytes	Muy grande
		double	Decimal doble	8 bytes	Muy grande
		char	Carácter simple	2 bytes	---
		boolean	Valor true o false	1 byte	---
	TIPOS OBJETO (con métodos, necesitan una invocación para ser creados)	Tipos de la biblioteca estándar de Java		String (cadenas de texto) Muchos otros (p.ej. Scanner, TreeSet, ArrayList...)	
		Tipos definidos por el programador / usuario		Cualquiera que se nos ocurra, por ejemplo Taxi, Autobus, Tranvia	
		arrays		Serie de elementos o formación tipo vector o matriz. Lo consideraremos un objeto especial que carece de métodos.	
		Tipos envoltorio o wrapper (Equivalentes a los tipos primitivos pero como objetos.)		Byte	
				Short	
				Integer	
				Long	
				Float	
				Double	
				Character	
				Boolean	

Tomado de: (Rodríguez, A. (2006). Tipos de datos JAVA. España: APR Aprender a Programar. Recuperado de:  
[https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=419:tipos-de-datos-java-tipos-primitivos-int-boolean-y-objeto-string-array-o-arreglo-variables-cu00621b&catid=68&Itemid=188](https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=419:tipos-de-datos-java-tipos-primitivos-int-boolean-y-objeto-string-array-o-arreglo-variables-cu00621b&catid=68&Itemid=188))

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

#### Actividad de Trabajo Autónomo

1. Escriba el código para recorrer un arreglo de 100 números en forma inversa, pero cada 10 elementos mostrar un mensaje de dónde va el conteo.
2. Resuelva los siguientes algoritmos:
  - Generar los números pares entre p y q
  - Generar los primeros z múltiplos de 3
  - Generar la suma de m primeros múltiplos de 7 más los n primeros múltiplos de 9
3. Escriba el código para crear una clase abstracta con 3 atributos
4. Cree una función que imprima cualquier tipo de arreglo (tipo de datos genéricos <T>), ya sea numérico, string, etc.

#### Actividad de Comprobación del Trabajo Autónomo

Se verificarán las respuestas a las preguntas de la sección de trabajo autónomo, recuerde que las respuestas son en idioma inglés.


#### Materiales, equipos e insumos a utilizar

##### Materiales, equipos e insumos proporcionados por la Universidad

Ítem	Cantidad
Un computador por estudiante	1
Conexión a Internet	1
Entorno Integrado de Desarrollo (IDE) Software Netbeans para programar en lenguaje Java.	1

##### Materiales del estudiante

Ítem	Cantidad
Un medio almacenamiento secundario para la realización de copia o backups	1
Conexión a Internet en el caso de trabajar en la casa	1
Entorno Integrado de Desarrollo (IDE) Software Netbeans para programar en lenguaje Java. <a href="https://netbeans.apache.org/download/archive/index.html">https://netbeans.apache.org/download/archive/index.html</a> <a href="https://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-3413139-esa.html">https://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-3413139-esa.html</a>	1

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

**Precauciones, nivel de riesgo y recomendaciones a considerar**

CLASIFICACIÓN DEL RIESGO	Muy alto	Medio
	Alto	Bajo
<b>FACTORES DE RIESGO</b>	<b>CÓMO MINIMIZAR LOS FACTORES DE RIESGO</b>	
El factor de riesgo para este laboratorio está clasificado como BAJ O, debido a que la eficacia del conjunto de las medidas preventivas existentes es alta.	En otras palabras, el riesgo está controlado	

**RECOMENDACIONES, CONSIDERACIONES PARA EL USO DE MATERIAL Y EPP**


En todo caso se deben seguir las siguientes recomendaciones para minimizar los riesgos propios de la actividad:

- *Identificar y conocer el protocolo de seguridad de laboratorios de informática.*
  - *No navegar en internet sin autorización del docente.*
  - *No ejecutar programas sin autorización del docente.*
  - *No instalar en los equipos Software de ninguna índole.*
  - *No trasladar equipos de cómputo de su módulo sin autorización del personal del área.*
  - *Cuidar sus objetos personales.*
  - *Cada alumno tiene como responsabilidad recibir las actividades de cada clase y apropiarse del material necesario para el desarrollo de las mismas.*
  - *Está prohibido el ingreso o consumo de alimentos, bebidas, chicle... dentro de la sala.*
  - *Está prohibido el uso e ingreso de dispositivos como celulares, parlantes y memorias USB sin autorización.*
  - *No conectar ni desconectar dispositivos como teclados, mouse o conexiones, en caso de anomalía avisar al profesor para realizar cambios o conexiones.*
  - *El trabajo debe hacerse en silencio, evitando las reuniones o interrumpiendo las actividades de otros estudiantes.*
  - *Cuide el buen funcionamiento del equipo que la ha sido asignado, evite cambiar configuraciones o intervenir los programas y propiedades del sistema operativo, el auxiliar de laboratorio es el único autorizado.*
  - *Todo dispositivo (teclado o mouse) que se pierda o se dañe con intención deberá ser repuesto.*
  - *No portar maletines o morrales, estos deben quedar depositados en los lockers destinados para ello. En caso de duda pida el respectivo candado con los auxiliares de cada laboratorio.*
  - *No rayar mesas, sillas, paredes y equipos, cuidar el aseo y orden de su puesto de trabajo.*
- Se prohíbe el ingreso o exploración de páginas no autorizadas y pornográficas, es causal de sanción y expulsión (vetado) de la sala de informática por varias sesiones.*

**CONSIDERACIONES ÉTICAS**

*Tener siempre presente el código de ética del ingeniero de software en especial el ítem relacionado con: Colegas: Cada ingeniero deberá apoyar y ser justos con los colegas, motivando a sus colegas sujetándose al código, ayudando también a su desarrollo profesional, reconocer los trabajos de otros y abstenerse a atribuirse de méritos indebidos, revisar los trabajos de manera objetiva, sincera y propiamente documentada. En caso de ser necesarias.*



	Nombre del Proceso: <b>GESTIÓN DE LABORATORIOS</b>	<b>CODIGO: LA-FM-001</b>
	Nombre del Documento: <b>FORMATO PRACTICAS DE LABORATORIOS</b>	<b>VERSION: 7</b>
		<b>FECHA: 15/junio/2022</b>

### Procedimiento y Metodología de la práctica


#### PROCEDIMIENTO A UTILIZAR.

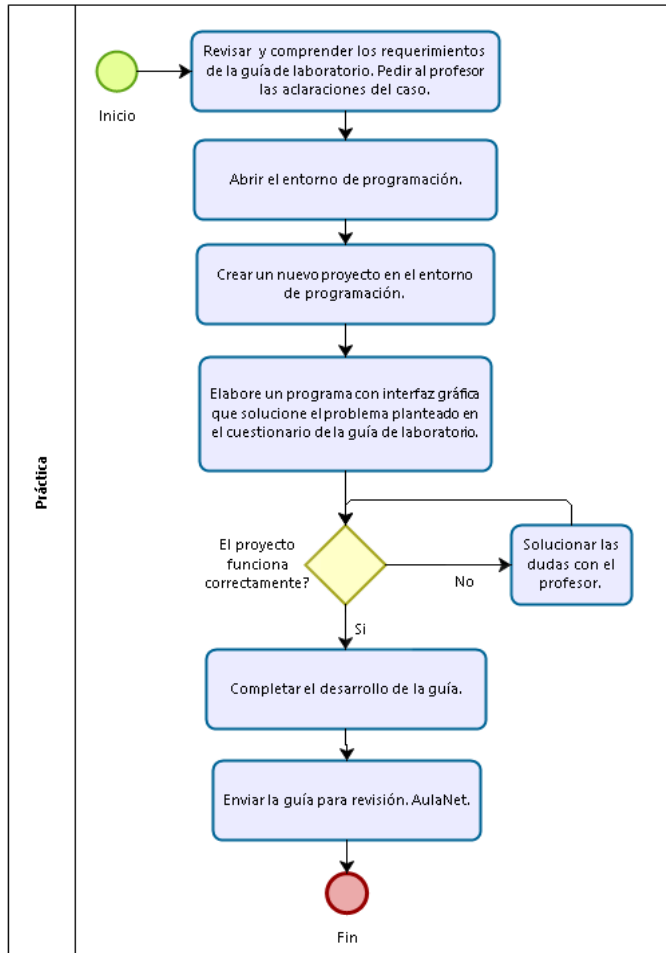
Realizar la **Actividad de trabajo independiente/autónomo**

Abrir la herramienta del entorno de trabajo **NetBeans**, Eclipse, VisualStudio.Net

***Realizar todos los ejercicios solicitados en la sección Introducción, estos ejercicios se deben solucionar en lenguaje Java utilizando la herramienta Netbeans.***

Verificar sus resultados siguiendo el siguiente diagrama:

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022




#### Criterios de Entrega – Informe de Laboratorio

-Formato de presentación docx y/o adjuntar los archivos en con la extensión de la aplicación Netbeans. Se debe comprimir el proyecto o proyectos que se construyó en lenguaje Java en la herramienta Netbeans, y también entregar todo el código en archivos .txt

-Uso de normas APA

En la siguiente tabla colocar por cada uno de los ejercicios solicitado su respectivo algoritmo en java adicionar también los pantallazos de las pruebas realizadas donde se evidencie que el programa ejecuto correctamente con diferentes casos de prueba (imágenes legibles) y el respectivo Código en Java.

Adjunte código fuente y pantallazos de ejecución de los programas solicitados

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

### Situaciones(10) en las que se aplica las estructuras de datos:

1. creacion de paginas de ventas
2. apps para ingreso de materias para universidades
3. boletines de colegios
4. nóminas de pago
5. tabla de puntaje en videojuegos
6. información económica de empresas
7. app de comunicación
8. apps para inventario
9. app para horario de buses
10. ap para viajes

### Algoritmo de Vehículos:

```
list = {
    19 : {},
    20 : {},
    21 : {},
    22 : {},
    23 : {}
}

def indata(list):
    s = int(input("ingrese el año al que desea ingresar un
auto:\t"))
    while s != None:
        if (s >= 19 and s <= 23) or (s >= 2019 and s <= 2023):
            for i in list:
                if s == i:
                    a = input("marca:\t")
                    list[i][a] = int(input("precio:\t"))
                    return list
            else:
                print("año invalido")
                return list

def pr(list):
    for i in list:
```



Nombre del Proceso:

GESTIÓN DE LABORATORIOS

CODIGO: LA-FM-001

Nombre del Documento:

FORMATO PRACTICAS DE LABORATORIOS

VERSION: 7

FECHA: 15/junio/2022

```
print("auto:\t" + str(i))
for a in list[i]:
    print(a + "\t" + str(list[i][a]))


def bara(list):
    for i in list:
        x = 0
        for a in list[i]:
            if x < list[i][a]:
                s = a
                x = list[i][a]

        print("año: "+i+"\nel auto mas barato fue: "+s+"\tprecio: "+x)

def prom(list):
    x = {
        19 : {},
        20 : {},
        21 : {},
        22 : {},
        23 : {}
    }
    for i in list:
        for a in list[i]:
            if list[i][a] > 30000000 and list[i][a] < 50000000:
                x[i][a] = list[i][a]

    print(x)

while True:
    print(
        "lista de opciones\n1) ingresar datos\n2) mostrar
datos\n3) mostrar los autos más baratos de cada año\n4) el
promedio de los autos que cuestan entre 30 y 50 millones"
```

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

```

)
s = int(input("ingrese su opcion:\t"))
if s == 1:
    list = indata(list)
elif s == 2:
    pr(list)
elif s == 3:
    bara(list)
elif s == 4:
    prom(list)
else:
    print("error, seleccione una opcion valida")


```

### Algoritmo Semana:

```

ventas = {
    "lunes":{
        1:None,
        2:None,
        3:None,
        4:None,
        5:None,
        6:None
    },
    "martes":{
        1:None,
        2:None,
        3:None,
        4:None,
        5:None,
        6:None
    },
    "miercoles":{
        1:None,
        2:None,
        3:None,


```

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

```

    4:None,
    5:None,
    6:None
  },
  "jueves":{
    1:None,
    2:None,
    3:None,
    4:None,
    5:None,
    6:None
  },
  "viernes":{
    1:None,
    2:None,
    3:None,
    4:None,
    5:None,
    6:None
  },
  "sabado":{
    1:None,
    2:None,
    3:None,
    4:None,
    5:None,
    6:None
  },
  "domingo":{
    1:None,
    2:None,
    3:None,
    4:None,
    5:None,
    6:None
  }

```

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

```

}

def main (ventas):
    print("opciones:\n1) ingresar datos\n2) ventas
superiores\n3) ventas deseadas")
    s = int(input("->"))

    if s == 1:
        ventas = setdata(ventas)
    elif s == 2:
        print(sup(ventas))
    elif s == 3:
        ventas = arre(ventas)
    else:
        print ("opcion incorrecta")

def setdata(ventas):
    t = int(input("ingrese el ID de su tienda:"))
    d = input("ingrese dia: (en minuscula y sin caracteres
especiales) ")

    if d in ventas:
        if t in ventas[d]:
            ventas[d][t] = float(input("total del dia : "))
            return ventas
        else:
            print("tineda invalida")
            return None
    else:
        print("dia invalido")
        return(ventas)

def sup(ventas):
    t = 0
    c = 0
    for i in ventas:

```



Nombre del Proceso:

**GESTIÓN DE LABORATORIOS**

**CODIGO: LA-FM-001**

Nombre del Documento:

**FORMATO PRACTICAS DE LABORATORIOS**

**VERSION: 7**

**FECHA: 15/junio/2022**

```
        for a in ventas[i]:
            t += ventas[i][a]
            c += 1

    t = t / c
    c = {}
    for i in ventas:
        for a in ventas[i]:
            if ventas[i][a] >= t:
                c[i][a] = ventas[i][a]

    return c

def arre(ventas):
    t = 0
    c = 0
    for i in ventas:
        for a in ventas[i]:
            t += ventas[i][a]
            c += 1


    t = t / c

    for i in ventas:
        for a in ventas[i]:
            if ventas[i][a] < t:
                ventas[i][a] = ventas[i][a] + (ventas[i][a] *
0.15)


while True:
    main(ventas)
```

**Criterios de Evaluación – Práctica de Laboratorio**




	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

RUBRICA DE EVALUACIÓN ESTRUCTURAS DE DATOS									
INDIVIDUAL			CRITERIOS DE EVALUACIÓN					NOTA	
Habilidad	Estudiante		0 – 1,5	1,6 - 2,9	3,0 - 3,9	4,0 - 4,5	4,6 - 5,0		
	1. Arlex Julian Cifuentes Ovalle								
	2.								
Competencia (aprender a hacer)	GRUPAL		%	0 – 1,5	1,6 - 2,9	3,0 - 3,9	4,0 - 4,5	4,6 - 5,0	NOTA
	P r o c e d i m e n t a l ( a p r e n d e r a h a c e r )	Organiza los resultados obtenidos a través del uso de gráficas, dibujos, tablas, mapas conceptuales. Documenta y organiza el código fuente y demás recursos utilizados para la solución del problema planteado.		No son precisos o no ayudan a la comprensión del tema.	Son organizados y algunas veces ayudan a la comprensión del tema.	Son organizados y ayudan al entendimiento del tema.	Son precisos y ayudan a la comprensión del tema.	Son organizados, precisos y ayudan al entendimiento del tema	
		Hace uso adecuado de los conceptos al momento de aplicarlos en la práctica experimental propuesta.	35%	Se limita a la recopilación de la información solicitada.	Hace referencia a la información en la práctica.	Hace relaciones básicas de la información con la práctica.	Analiza la información relacionada con la práctica.	Infiere la información obtenida y la relaciona con su realidad.	
	Competencia (aprender a conocer)	Utiliza correctamente los tipos de datos, declaración de variables, funciones, estructuras de control condicional y repetitivas para la solución de un problema.		Se le dificulta la declaración de variables, el uso de funciones y de estructuras de control.	Declara variables. Se le dificulta el uso de funciones y de estructuras de control.	Declara variables, utiliza funciones. Se le dificulta el uso de estructuras de control.	Declara variables, utiliza funciones y estructuras de control.	Utiliza correctamente la declaración de variables, el uso de funciones y de estructuras de control.	
		Comprende el concepto y uso de las estructuras de datos en ingeniería.	35%	No comprende el concepto, uso y aplicaciones de las estructuras de datos en ingeniería.	Comprende el concepto de estructuras de datos. No identifica su uso y aplicaciones.	Comprende el concepto de estructuras de datos y algunas aplicaciones de las mismas.	Comprende el concepto, uso y aplicaciones de las estructuras de datos en ingeniería.	Comprende el concepto, uso y aplicaciones de las estructuras de datos en ingeniería y en su profesión.	
Soc	Desarrolla habilidades de trabajo en equipo, priorizando la toma de decisiones y la		100%	No hace parte del trabajo	Parcialmente hace parte del trabajo	Hace parte del trabajo propuesto por	Participa en el trabajo propuesto por	Participa activamente el trabajo propuesto	

	Nombre del Proceso:	CODIGO: LA-FM-001
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	VERSION: 7
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	FECHA: 15/junio/2022

i o a f e c t i v a ( a p r e n d e r a s e r )	escucha de diferentes propuestas.		propuesto por el equipo.	propuesto por el equipo.	el equipo de acuerdo a parámetros básicos.	el equipo de manera responsable y puntual.	por el equipo de manera responsable y puntual.		
	Cuida, respeta y exige respeto frente a la interacción con sus pares y docentes.		Frecuentemente reprocha el trabajo de sus pares y docente, y justifica sus carencias en el trabajo en grupo.	A veces muestra una actitud favorable frente a la clase y se limita a responder por las condiciones básicas del trabajo.	Muestra una actitud favorable frente a la clase e interactúa ocasionalmente con sus pares y docente.	Muestra una buena actitud frente a la clase e interactúa con sus pares y docente.	Muestra una sobresaliente actitud frente a la clase e interactúa frecuentemente con sus pares y docente.		
C o m u n i c a t i v a ( a p r e n d e r a c o n v i v i r )	Practica el uso de lenguaje escrito como medio de identificación y diferenciación en la elaboración de informes.		No construye el informe de laboratorio.	Construye el informe de laboratorio de manera incompleta.	Construye el informe de laboratorio de acuerdo a los requerimientos mínimos.	Construye de buena manera el informe de laboratorio.	Construye de manera sobresaliente el informe de laboratorio.		
	Utiliza lenguaje técnico para referirse a los diferentes conceptos que relaciona en la práctica experimental.	100%	No hace uso de un lenguaje técnico apropiado para la práctica de laboratorio.	Ocasionalmente hace uso de un lenguaje técnico apropiado para la práctica de laboratorio.	Hace uso de un lenguaje técnico apropiado para la práctica de laboratorio.	Hace un buen uso del lenguaje técnico apropiado para la práctica de laboratorio.	Hace un uso sobresaliente del lenguaje técnico apropiado para la práctica de laboratorio.		
I n v e s t i g a t i v a	Realiza la búsqueda bibliográfica en fuentes confiables que permitan dar respuesta a las situaciones problema evidenciadas en la consulta previa y en el informe de laboratorio.	100%	Las fuentes de información son pocas o ausentes. Si las usa son poco confiables y no contribuyen a la construcción del eje central.	Las fuentes de información son restringidas. No son actualizadas y contienen información poco relevante.	Las fuentes de información son relevantes e informativas. Presenta los parámetros aceptables por el docente.	Las fuentes de información son variadas e informativas. Adicionalmente son fiables y contribuyen al tema.	Las fuentes de información son variadas y pertinentes. Además están actualizadas y contienen información relevante al tema.		

	Nombre del Proceso:	<b>CODIGO: LA-FM-001</b>
	<b>GESTIÓN DE LABORATORIOS</b>	
	Nombre del Documento:	<b>VERSION: 7</b>
	<b>FORMATO PRACTICAS DE LABORATORIOS</b>	<b>FECHA: 15/junio/2022</b>

	Consulta el estado del arte sobre computación paralela y elabora un informe sobre API para computación paralela.	No realiza la consulta ni el informe.	Realiza la consulta pero no presenta el informe.	Presenta el informe de manera incompleta.	Presenta el informe.	Cumple con los requerimientos del informe.		
--	--	---------------------------------------	--	---	----------------------	--	--	--

#### Palabras Clave

#### ESTRUCTURAS DATOS

#### Bibliografía Recomendada

Deitel, H. M.(1998). Cómo programar en Java. México : Prentice-Hall Hispanoamericana. (Colección Biblioteca UMB)

Hernández, R. (2001), Estructuras de datos y algoritmos. Prentice-Hall, 283 p.

Howard, M., LeBlanc, D. (2010). 24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them. New York: The McGraw-Hill Companies (Recuperado Base de Datos Access Engineering)

Joyanes A. (1999), Estructura de datos : libro de problemas. McGraw-Hill / Interamericana, 550 p.

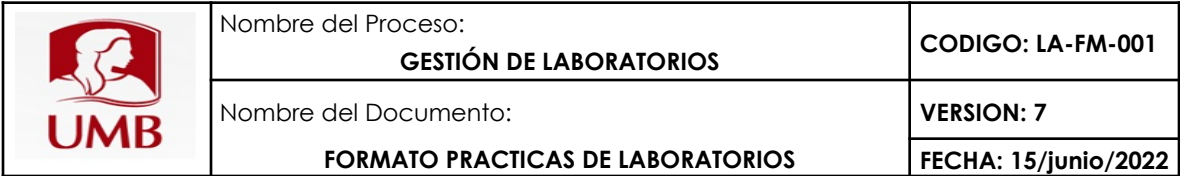
Joyanes, A. L., Rodríguez, B. L., & Fernández, A. M. (2003). Fundamentos de programación: libro de problemas. algoritmos, estructuras de datos y objetos (2a. ed.). (Recuperado Base de Datos E-Libro)

Vélez, S. J., Peña, A. A., & Gortazar, B. P. (2011). Diseñar y programar, todo es empezar: una introducción a la programación orientada a objetos usando uml y java.(Recuperado Base de Datos E-Libro)

#### REFERENCIAS DE BASES DE DATOS

- Olier Quiceno, A. J., Gómez Salgado, A. A., & Caro Pineros, M. F. (2017). Design and Implementation of a Teaching Tool for Introduction to object-oriented programming. IEEE Latin America Transactions, 15(1), 97-102 (Recuperado Base de Datos EBSCO)
- Vijayasathy L. R. Butler C. W. (2016), Choice of Software Development Methodologies: Do Organizational, Project, and Team Characteristics Matter?“, IEEE Software (Volume: 33, Issue: 5) (Recuperado Base de Datos IEEE)
- Pressman, R. S., aCampos Olgún, V., & Enríquez Brito, J. (2010). Ingeniería del software: un enfoque práctico. Séptima edición. (Recuperado Base de Datos eBooks 7-24)

Tema	Subtema	Referente bibliográfico
Introducción a las estructuras de datos: variables, condicionales y bucles (repaso), programación	Definición y uso de las estructuras de datos.	Joyanes A. (1999), Estructura de datos : libro de problemas. McGraw-Hill / Interamericana, 550 p.
		Hernández, R. (2001), Estructuras de datos y algoritmos. Prentice-Hall, 283 p.
	Aplicaciones de las estructuras de	Howard, M., LeBlanc, D. (2010). 24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them. New



n orientada a objetos (repaso)	datos en ingeniería de software.	York: The McGraw-Hill Companies (Recuperado Base de Datos Access Engineering)
		Pressman, R. S., aCampos Olguín, V., & Enríquez Brito, J. (2010). Ingeniería del software: un enfoque práctico. Séptima edición. (Recuperado Base de Datos eBooks 7-24)
	Conceptos básicos de programación (repaso)	Joyanes, A. L., Rodríguez, B. L., & Fernández, A. M. (2003). Fundamentos de programación: libro de problemas. algoritmos, estructuras de datos y objetos (2a. ed.). (Recuperado Base de Datos E-Libro)

Control de cambios		
Fecha de Actualización	Descripción	Participantes
12/12/2023		Hugo Ortiz