



Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software.

Nombre de la Práctica Arreglos unidimensionales

Numero de Práctica 3

Nombre de la carrera Ingeniería de Software

Nombre de la materia Lab. Estructuras de Datos

Nombre del alumno Jesús Manuel Juárez Pasillas

Nombre del docente Aldonso Becerra Sánchez

Fecha: 24/08/2021

Práctica 3: Arreglos unidimensionales

Introducción:

Los arreglos unidimensionales son estructuras de datos lineales en los cuales se puede almacenar una cierta cantidad de elementos definida al momento de crear la instancia. Estos arreglos son muy útiles al utilizar pocos elementos a guardar. Manejar información guardada en arreglos permite resolver una gran cantidad de problemas puestos por un programa, en cuanto a la información que se requiere almacenar y manipular con diferentes operaciones y que los datos almacenados interactúen entre sí.

Desarrollo:

En el desarrollo de esta práctica se estará haciendo uso de la clase ArregloDatos creado y complementado la practica anterior.

En esta práctica se agregaron los siguientes clases y métodos:

Ejido: Esta clase manipula un arreglo el cual estará almacenando datos de tipo Campesino. En ella se agregaron los siguientes métodos:

```
- Agrega un campesino al arreglo campesinos.
public boolean agregarCampesino(Campesino campesino);

- Obtiene el promedio de las cosechas de un campesino en particular.
public Double promedioCosechaCampesino(String nombreCampesino,String cosecha);

- Obtiene el campesino que obtuvo la menor cantidad de cosecha indicada en el año indicado.
public Campesino menosCosecha(String nombreCos,int year);

- Indica el campesino que más le favoreció las siembras en un año en particular.
public Campesino favorecido(int year);

- Evalúa que cosecha convino más a partir del año.
public Cosecha cosechaMasConvinoAnio(int year);

- Obtiene el año en el que se produjo más producto de cierto tipo, de un campesino en particular.
public Anio anioMejorProdujo(String nomCampesino,String producto);

- Obtiene la persona que menos ha cosechado en el último año.
public Campesino menosCosechaActual();
```

- Recibe como argumento un arreglo con el cual obtendrá los campesinos que se desean saber la suma de un producto.
`public Double cantidadTonelCampesinos(ArregloDatos campes, String producto);`
- Obtiene el campesino que produce menos de la cantidad indicada de toneladas de productos en general.
`public Campesino produceMenosQue(double cantidad);`
- Manda llamar el metodo produceMasQue de cada campesino.
`public Cosecha produceMasQue(double cantidad);`

Campesino: Clase que simula ser un campesino el cual tendrá un nombre y un arreglo de años. En esta clase se agregaron los siguientes métodos:

- Obtiene el nombre del campesino.
`public String getNombre();`
- Cambia el nombre del campesino.
`public void setNombre(String nombre);`
- Obtiene el arreglo que contiene los años.
`public ArregloDatos getAnios();`
- Cambia el arreglo de años.
`public void setAnios(ArregloDatos anios);`
- Agrega un año al arreglo años.
`public boolean agregarAnio();`
- Obtiene el promedio de las cosechas en general del campesino.
`public Double promedioCosecha(String nombreCosecha);`
- Obtiene el total de toneladas producidas de un solo producto.
`public Double cantidadTonel(String nomCosecha);`
- Evalúa si el promedio de toneladas de un producto son mayores a la cantidad indicada.
`public Cosecha produceMasQue(double cantidad);`

Anio: Esta clase simula ser un año el cual almacena un arreglo de cosechas del mismo año. A esta clase se le agregaron los siguientes métodos:

- Obtiene el año.
`public int getAnio();`
- Cambia el valor de año.
`public void setAnio(int anio);`
- Obtiene las cosechas del año.
`public ArregloDatos getCosechas();`

```

- Cambia las cosechas del año.
  public void setCosechas(ArregloDatos cosechas);

- Crea las cosechas solicitadas.
  public void agregarCosechas();

- Agrega una cosecha al arreglo cosechas.
  public boolean agregarCosecha(Cosecha cosecha);

```

Cosecha: Esta clase simula ser una cosecha la cual guardara el nombre de la cosecha y las toneladas. En esta clase se agregaron los siguientes métodos:

```

- Obtiene el nombre de la cosecha.
  public String getNombre();

- Cambia el nombre de la cosecha.
  public void setNombre(String nombre);

- Obtiene las toneladas de la cosecha.
  public double getToneladas();

- Cambia las toneladas de la cosecha.
  public void setToneladas(double toneladas);

```

Todas estas clases cuentan con su respectiva interface la cual se encuentra a la misma altura que estas clases ("src/registros/ejido") y las cuales se identifican como interface por una "I" antes del nombre de la clase que la implementa.

Nota: Toda la documentación esta agregada en la carpeta "doc" dentro de la carpeta del proyecto ("edylab_2021_3/doc").

Capturas del programa funcionando:

Pide los años y las toneladas para cada producto.

```

Ingrese el año (Campesino1): 2019
Ingresa las toneladas para el Frijol: 12
Ingresa las toneladas para el Maiz: 11
Ingresa las toneladas para el Chile: 12
Ingresa las toneladas para el Zanahoria: 12
Ingresa las toneladas para el Ajo: 11
Ingresa las toneladas para el Cebolla: 11
Ingrese el año (Campesino1): 2020
Ingresa las toneladas para el Frijol: 15
Ingresa las toneladas para el Maiz: 14
Ingresa las toneladas para el Chile: 16
Ingresa las toneladas para el Zanahoria: 15
Ingresa las toneladas para el Ajo: 15
Ingresa las toneladas para el Cebolla: 4
Ingrese el año (Campesino2): 2019
Ingresa las toneladas para el Frijol: 12
Ingresa las toneladas para el Maiz: 16
Ingresa las toneladas para el Chile: 17
Ingresa las toneladas para el Zanahoria: 12
Ingresa las toneladas para el Ajo: 17
Ingresa las toneladas para el Cebolla: 17

```

Llama los métodos.

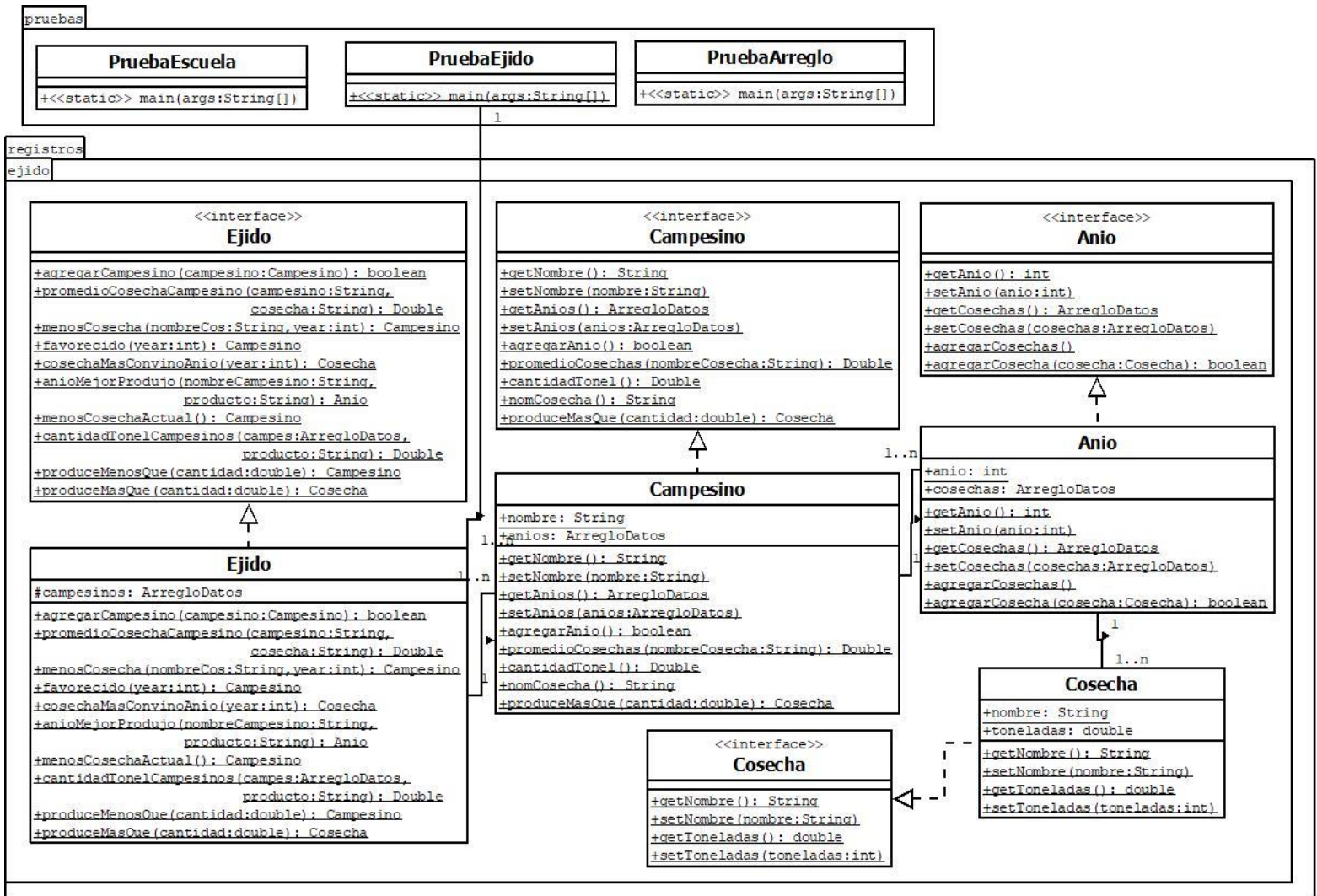
```

A. El promedio de toneladas producidas por un campesino
en particular de un producto en especifico.
Promedio: 2.0
B. El campesino produjo menos toneladas de frijol
en un año en particular.9
Campesino: Campesino3
C. Indique cuál campesino fue el más favorecido
en sus siembras en un año en particular
Campesino favorecido: Campesino4
D. ¿Qué producto convino más en la
siembra en un año en particular?
Cosecha: Frijol
E. ¿Qué año produjo mejores cantidades de producto
"Cebolla" para el campesino "Campesino5"?Año: 2019
G. ¿Qué persona es la que menos
ha cosechado en este año?
Campesino: null
H. ¿Cuántas toneladas producen de maíz
Campesino1, Campesino4 y Campesino7?
Toneladas: 103.0
I. ¿Qué producto produce más de
50 toneladas en promedio por año?
Cosecha: null
J. ¿Qué campesino produce menos de
40 toneladas de productos por año?
Campesino: Campesino3

```

Código agregado:

Todas las clases dentro del paquete ejido son nuevas al igual que los métodos que están dentro de estas clases, también la clase PruebaEjido es nueva.



Pre-evaluación:

Pre-Evaluación para prácticas de Laboratorio de Estructuras de Datos	PRE-EVALUACIÓN DEL ALUMNO
CUMPLE CON LA FUNCIONALIDAD SOLICITADA.	No
DISPONE DE CÓDIGO AUTO-DOCUMENTADO.	Sí
DISPONE DE CÓDIGO DOCUMENTADO A NIVEL DE CLASE Y MÉTODO.	Sí
DISPONE DE INDENTACIÓN CORRECTA.	Sí
CUMPLE LA POO.	Sí
DISPONE DE UNA FORMA FÁCIL DE UTILIZAR EL PROGRAMA PARA EL USUARIO.	Sí
DISPONE DE UN REPORTE CON FORMATO IDC.	Sí
LA INFORMACIÓN DEL REPORTE ESTÁ LIBRE DE ERRORES DE ORTOGRAFÍA.	Sí
SE ENTREGÓ EN TIEMPO Y FORMA LA PRÁCTICA.	No
INCLUYE LA DOCUMENTACIÓN GENERADA CON JAVADOC.	Sí
INCLUYE EL CÓDIGO AGREGADO EN FORMATO UML.	Sí
INCLUYE LAS CAPTURAS DE PANTALLA DEL PROGRAMA FUNCIONANDO.	Sí
LA PRÁCTICA ESTÁ TOTALMENTE REALIZADA (ESPECIFIQUE EL PORCENTAJE COMPLETADO).	90%
Observaciones:	

Conclusión:

Usar los arreglos para la manipulación de datos nos permite manipular un conjunto de datos relacionados con los que podemos hacer operaciones y obtener un resultado querido, como lo es obtener el mejor, peor, un promedio, etc.

Los datos que se almacenan en un arreglo pueden ser mas arreglos para poder manipular mas datos relacionados dentro de un dato almacenado en un arreglo.

Varios de los métodos agregados a la clase ArregloDatos nos permitió manipular los datos que se almacenaron dentro de estos arreglos.