



**Universidad Autónoma de Zacatecas**

**Unidad Académica de Ingeniería Eléctrica**

**Programa Académico de Ingeniería de Software.**

**Nombre de la Práctica      Repaso de POO**

**Numero de Práctica          2**

**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: 18/08/2021**

## Práctica 2: Arreglos

### Introducción:

El principal objetivo de los arreglos es almacenar datos en una sola variable. Los arreglos tienen que ser definidos con el número de posiciones que se quiera tener y esta longitud nunca cambiara, sin importar que todas las posiciones se utilicen o no.

Los datos que se almacenan son muy fáciles de obtener ya que la estructura del arreglo permite tener un índice para cada posición y de esta forma acceder al elemento deseado.

### Desarrollo:

En el desarrollo de esta práctica se estarán agregando varios métodos para manipular los datos almacenados, así como a el mismo arreglo.

Toda la documentación esta agregada en la carpeta “doc” dentro de la carpeta del proyecto (“edylab\_2021\_2/doc”).

### Capturas del programa funcionando:

```
Arreglo1:
A
G
D
S
A

Arreglo2:
A
G
S
S
r

¿arreglo1 es igual a arreglo2? false
Buscar en arreglo2 S
Cambiar A->S en arreglo2 true
S
G
S
S
r
```

```
Eliminar arreglo2 de arreglo4 true
r
S
r
L
S
```

```
Clonar arreglo4
r
S
r
L
S
A
A
A
```

```
Cambiar G->L en arreglo2 true
S
L
S
S
r

Cambiar arregloDatos false
Buscar s
Eliminar la primera posicion en arreglo2S

L
S
S
S
r
Eliminar ultimo dato r

L
S
S

Vaciar arreglo1
null
null
null
null
null
```

```
Rellenar arreglo4 con 3 A
r
S
r
L
S
A
A
A
```

```
Sub lista
r
L
S

Redimensionar arreglo4 a 5 posiciones
r
S
r
L
S
```

```
Lista 4:
A
G
S
S
r

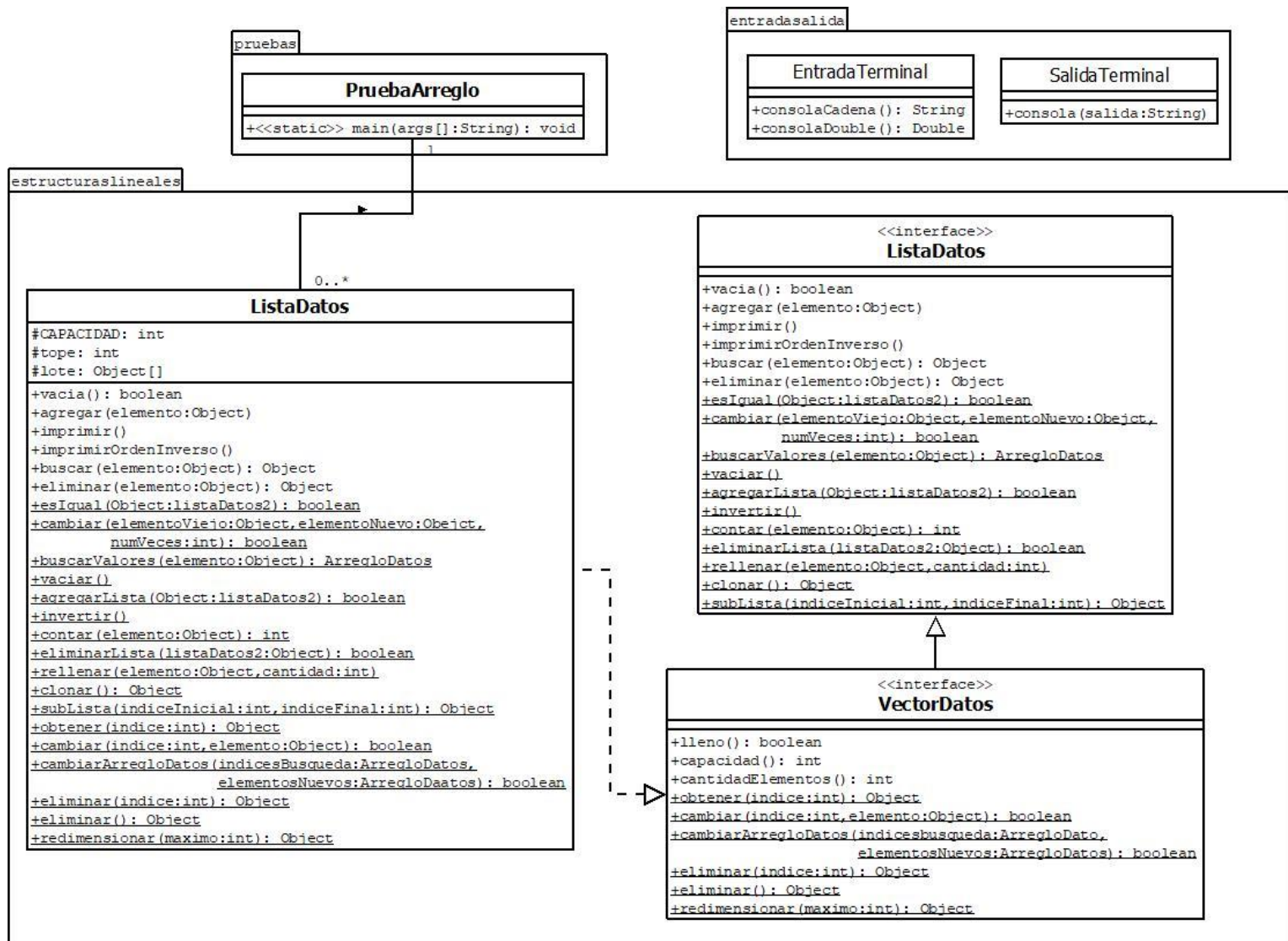
Agregar lista 2 a arreglo 4true
A
G
S
S
r
L
S

Invertir arreglo 4
S
L
r
S
r
L
S

Contar S en arreglo4: 2
```

## Código agregado:

Todos los métodos subrayados son nuevos y es el código que se agregó.



## Pre-evaluación:

Pre-Evaluación para prácticas de Laboratorio de Estructuras de Datos	PRE-EVALUACIÓN DEL ALUMNO
CUMPLE CON LA FUNCIONALIDAD SOLICITADA.	Sí
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.	Sí
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).	100%
Observaciones:	

## Conclusión:

Los arreglos son muy útiles cuando se trata de almacenar grupos de datos en una sola variable con la cual acceder a cada uno de los datos almacenados, mientras los datos no sean una cantidad enorme y se conozca la cantidad de datos que se quieren almacenar y que esta no cambie.

Es muy importante usar el tipo de dato "Object", ya que sin este sería mucho más complicado ya que solo se admitiría un tipo de dato en los arreglos. Además, que sería muy extenso poder hacer todas las funciones antes mencionadas para cada tipo de dato que se esté manejando.