



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
**FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN**

ALUMNO:

**PAZ MALDONADO CARLOS SAÚL**

NOMBRE DEL PROFESOR:

**HERNANDEZ CABRERA JESUS**

NOMBRE DE LA MATERIA:

**ESTRUCTURA DE DATOS**

FECHA DE ENTREGA:

**23 de agosto del 2024**

TAREA NO. 2

# CÓDIGOS

## Clase ConjuntoADT

```
package fesaragon.estructuradatos;

public class ConjuntoADT<T> {
    private T[] conjunto;
    private int longitud;

    public ConjuntoADT() {
        this.conjunto = (T[]) new Object[0];
        this.longitud = 0;
    }

    public int longitud() {
        return this.longitud;
    }

    public boolean contiene(T elemento) {
        for (T aux : this.conjunto) {
            if (aux.equals(elemento)) {
                return true;
            }
        }
        return false;
    }

    public void agregar(T elemento) throws
    ExcepcionConjuntoADT {
        if (contiene(elemento)) {
            throw new ExcepcionConjuntoADT("El elemento
ya se encuentra en el conjunto");
        }

        T[] aux = (T[]) new Object[longitud + 1];
        for (int i = 0; i < this.conjunto.length; i++)
        {
            aux[i] = this.conjunto[i];
        }
    }
}
```

```

    }
    aux[longitud] = elemento;
    this.conjunto = aux;
    this.longitud++;
}

    public void eliminar(T elemento) throws
ExcepcionConjuntoADT {
    if (!contiene(elemento)) {
        throw new ExcepcionConjuntoADT("No se
encuentra el elemento para eliminarlo");
    }
    T[] aux = (T[]) new Object[longitud - 1];
    int j = 0; // Es para controlar el indice del
arreglo aux
    for (int i = 0; i < this.conjunto.length; i++)
    {
        if (!conjunto[i].equals(elemento)) {
            aux[j++] = conjunto[i];
        }
    }
    this.conjunto = aux;
    this.longitud--;
}

    public boolean equals(ConjuntoADT<T> conjunto2) {
        if (this.longitud != conjunto2.longitud){
            return false;
        }
        for (T elemento1 : this.conjunto) {
            if (!conjunto2.contiene(elemento1)) {
                return false;
            }
        }

        return true;
    }

    public boolean esSubConjunto(ConjuntoADT<T>

```

```

conjunto2) throws ExcepcionConjuntoADT{
    if (conjunto2.longitud > this.longitud ){
        throw new ExcepcionConjuntoADT("El conjunto
con el que estas comparando tiene mayores elementos,
por ende no puede ser un subconjunto");
    }
    for (T elemento : this.conjunto) {
        if (!conjunto2.contiene(elemento)) {
            return false;
        }
    }
    return true;
}

    public ConjuntoADT<T> union(ConjuntoADT<T>
conjunto2) throws ExcepcionConjuntoADT {
        ConjuntoADT<T> nuevoConjunto = new
ConjuntoADT<>();
        for (T elemento : this.conjunto) {
            nuevoConjunto.agregar(elemento);
        }
        for (T elemento : conjunto2.conjunto) {
            if (!nuevoConjunto.contiene(elemento)) {
                nuevoConjunto.agregar(elemento);
            }
        }
        return nuevoConjunto;
    }

    public ConjuntoADT<T> interseccion(ConjuntoADT<T>
conjunto2) throws ExcepcionConjuntoADT {
        ConjuntoADT<T> nuevoConjunto = new
ConjuntoADT<>();
        for (T elemento : this.conjunto) {
            if (conjunto2.contiene(elemento)) {
                nuevoConjunto.agregar(elemento);
            }
        }
        return nuevoConjunto;
    }

```

```

    }

    public ConjuntoADT<T> diferencia(ConjuntoADT<T>
conjunto2) throws ExcepcionConjuntoADT {
        ConjuntoADT<T> nuevoConjunto = new
ConjuntoADT<>();
        for (T elemento : this.conjunto) {
            if (!conjunto2.contiene(elemento)) {
                nuevoConjunto.agregar(elemento);
            }
        }
        return nuevoConjunto;
    }
    public void imprimir() {
        System.out.print("{ ");
        for (T elemento : conjunto) {
            System.out.print(elemento + " ");
        }
        System.out.println("}");
    }
}

```

## Clase de Excepciones

```

package fesaragon.estructuradatos;

public class ExcepcionConjuntoADT extends Exception{
    public ExcepcionConjuntoADT(String message) {
        super(message);
    }
}

```

## Clase MAIN

```
package fesaragon.estructuradatos;

import static java.lang.StringTemplate.STR;

public class Main {
    public static void main(String[] args) {
        try {
            ConjuntoADT<Integer> conjunto1 = new
ConjuntoADT<>();
            ConjuntoADT<Integer> conjunto2 = new
ConjuntoADT<>();
            conjunto1.agregar(1);
            conjunto1.agregar(2);
            conjunto1.agregar(3);
            conjunto2.agregar(3);
            conjunto2.agregar(4);
            conjunto2.agregar(5);
            System.out.println("\n\nVALORES DEL
CONJUNTO 1: ");
            conjunto1.imprimir();
            System.out.println("VALORES DEL CONJUNTO 2:
");
            conjunto2.imprimir();
            //Eliminar elemento
            System.out.println("\n");
            System.out.println("Eliminacion del numero
1 en el conjunto 1: ");
            conjunto1.eliminar(1);
            conjunto1.imprimir();
            System.out.println();
            //Existe un elemento 2 en el conjunto 1
            String existeElemento = STR."El conjunto 1
contiene el número 2: \{conjunto1.contiene(2)}\n";
            System.out.println(existeElemento);
            // ConjuntoADT función: unión, intersección
y diferencia
```

```

        System.out.println("Elementos de los
conjuntos antes de union, interseccion y diferencia
\nValores del conjunto 1: ");
        conjunto1.imprimir();
        System.out.println("Valores del conjunto 2:
");
        conjunto2.imprimir();
        System.out.println("\n");
        ConjuntoADT<Integer> union =
conjunto1.union(conjunto2);
        System.out.println(STR."Longitud de
conjunto union: \{union.longitud()} \nValores del
conjunto union: ");
        union.imprimir();
        ConjuntoADT<Integer> interseccion =
conjunto1.interseccion(conjunto2);
        System.out.println(STR."Longitud de
conjunto intersección: \{interseccion.longitud()}
\nValores del conjunto intersección: ");
        interseccion.imprimir();

        ConjuntoADT<Integer> diferencia =
conjunto1.diferencia(conjunto2);
        System.out.println(STR."Longitud de
conjunto diferencia: \{diferencia.longitud()} \nValores
del conjunto diferencia: ");
        diferencia.imprimir();

    } catch (ExcepcionConjuntoADT e) {
        System.out.println(e.getMessage());
    }
}
}

```

## Capturas de ejecución

```
C:\Users\temoc\.jdk\openjdk-22.0.2\bin\java.exe --enable-preview "-java
```

```
VALORES DEL CONJUNTO 1:
```

```
{ 1 2 3 }
```

```
VALORES DEL CONJUNTO 2:
```

```
{ 3 4 5 }
```

```
Eliminacion del numero 1 en el conjunto 1:
```

```
{ 2 3 }
```

```
El conjunto 1 contiene el número 2: true
```

```
Elementos de los conjuntos antes de union, interseccion y diferencia
```

```
Valores del conjunto 1:
```

```
{ 2 3 }
```

```
Valores del conjunto 2:
```

```
{ 3 4 5 }
```

```
Longitud de conjunto union: 4
```

```
Valores del conjunto union:
```

```
{ 2 3 4 5 }
```

```
Longitud de conjunto intersección: 1
```

```
Valores del conjunto intersección:
```

```
{ 3 }
```

```
Longitud de conjunto diferencia: 1
```

```
Valores del conjunto diferencia:
```

```
{ 2 }
```

```
Process finished with exit code 0
```



## Observaciones:

Este código fue realizado con el jdk 22.0.2, también, se encuentran habilitadas las características de vista previa.

**Project**  
Default settings for all modules. Configure these parameters for each module on the module page as needed.

Name:

SDK:  Edit

Language level:  ▼

Compiler output:  📁

Used for module subdirectories, Production and Test directories for the corresponding sources.

Están habilitadas con el objetivo de utilizar las Template Strings en Java.