



Hernández Rubio Aarón Isaí

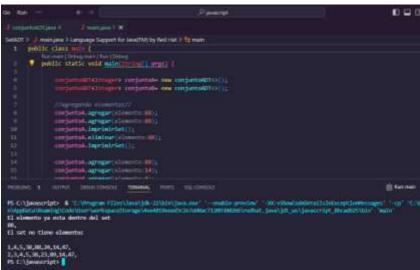
Estructura de Datos

Grupo:1360(2025-1)

Tarea 2: Implementación de conjuntoADT

23 de agosto de 2024

```
Description of the property of
```



```
Setton ( per per proper proper to maching by red ton 3 former 3 former 3 former 2 former 2 former 2 former 2 former 2 former 2 former 3 fo
```

```
| Images | I
```

CODIGO:

```
import java.util.HashSet;

public class conjuntoADT<T>{
    private HashSet<T> set = new HashSet<>();

public conjuntoADT(){
    }
    public conjuntoADT(HashSet<T> setPrev){
        this.set=setPrev;
}
```

```
public int longitud(){
   return this.set.size();
public boolean contiene(T elemento){
    if(this.set.contains(elemento)){
        return true;
    }else{
       return false;
public void agregar(T elemento){
    if(this.contiene(elemento)==true){
        System.out.println("El elemento ya esta dentro del set");
    }else{
        set.add(elemento);
public void eliminar(T elemento){
    if(this.contiene(elemento) == true){
        this.set.remove(elemento);
        System.out.println("El elemento no esta dentro del set");
public boolean equals(HashSet<T> otroConjunto){
    if(this.set.equals(otroConjunto)){
        return true;
    }else{
        return false;
public boolean esSubconjunto(HashSet<T> otroConjunto){
    if(otroConjunto.containsAll(this.set)){
        return true;
    }else{
        return false;
```

```
public HashSet<T> union(HashSet<T> otroConjunto){
   HashSet<T> setAux = new HashSet<>();
    setAux.addAll(this.set);
   setAux.addAll(otroConjunto);
   return setAux;
public HashSet<T> interseccion(HashSet<T> otroConjunto){
   HashSet<T> setAux= new HashSet<>(this.set);
    setAux.retainAll(otroConjunto);
   return setAux;
public HashSet<T> diferencia(HashSet<T> otroConjunto){
    HashSet<T> setAux= new HashSet<>(this.set);
    setAux.removeAll(otroConjunto);
    return setAux;
public HashSet<T> getSet(){
   return this.set;
public void imprimirSet(){
    if(this.longitud()==0){
        System.out.println("El set no tiene elementos");
    for(T elemento : this.set) {
        System.out.print(elemento + ",");
   System.out.println();
```

```
import java.util.HashSet;

public class main {
    public static void main(String[] args) {

        conjuntoADT<Integer> conjuntoA= new conjuntoADT<>();
        conjuntoADT<Integer> conjuntoB= new conjuntoADT<>();
```

```
//agregando elementos//
conjuntoA.agregar(88);
conjuntoA.agregar(88);
conjuntoA.imprimirSet();
conjuntoA.eliminar(88);
conjuntoA.imprimirSet();
conjuntoA.agregar(88);
conjuntoA.agregar(14);
conjuntoA.agregar(4);
conjuntoA.agregar(1);
conjuntoA.agregar(5);
conjuntoA.agregar(24);
conjuntoA.agregar(38);
conjuntoA.agregar(47);
conjuntoB.agregar(5);
conjuntoB.agregar(4);
conjuntoB.agregar(38);
conjuntoB.agregar(47);
conjuntoB.agregar(23);
conjuntoB.agregar(2);
conjuntoB.agregar(14);
conjuntoB.agregar(3);
conjuntoB.agregar(89);
if(conjuntoA.contiene(1)){
    System.out.println("El conjuntoA si contiene ese numero");
}else{
    System.out.println("El conjuntoA no contiene ese numero");
if(conjuntoB.contiene(1)){
    System.out.println("El conjuntoBsi contiene ese numero");
    System.out.println("El conjuntoB no contiene ese numero");
HashSet<Integer> conjuntoUnion=conjuntoA.union(conjuntoB.getSet());
```

```
conjuntoADT<Integer> conjuntoUnionADT= new
conjuntoADT<>(conjuntoUnion);
        System.out.println("Opereación Unión:");
        conjuntoUnionADT.imprimirSet();
        HashSet<Integer>
conjuntoInterseccion=conjuntoA.interseccion(conjuntoB.getSet());
        conjuntoADT<Integer> conjuntoInterseccionADT= new
conjuntoADT<>(conjuntoInterseccion);
        System.out.println("Opereación Intersección:");
        conjuntoInterseccionADT.imprimirSet();
conjuntoDiferencia=conjuntoA.diferencia(conjuntoB.getSet());
        conjuntoADT<Integer> conjuntoDiferenciaADT= new
conjuntoADT<>(conjuntoDiferencia);
        System.out.println("Opereación Diferencia:");
        conjuntoDiferenciaADT.imprimirSet();
        if(conjuntoA.equals(conjuntoB.getSet())){
            System.out.println("los conjuntos son iguales");
        }else{
            System.out.println("los conjuntos no son iguales");
        if(conjuntoA.esSubconjunto(conjuntoB.getSet())){
            System.out.println("El conjunto A es subconjunto del conjunto
B");
        }eLse{
            System.out.println("El conjunto A no es subconjunto del conjunto
```