Universidad Autónoma de México

Facultad de Estudios Superiores Aragón

Martínez González Ander Santiago

Tarea 2: Implementación del ConjuntoADT

Estructuras de Datos

Tarea 2.py > ConjuntoADT

```python
class ConjuntoADT:
    def __init__(self):
        self.elementos = []

    def agregar(self, elemento):
        if elemento not in self.elementos:
            self.elementos.append(elemento)

    def eliminar(self, elemento):
        if elemento in self.elementos:
            self.elementos.remove(elemento)

    def contiene(self, elemento):
        return elemento in self.elementos

    def union(self, otro_conjunto):
        nuevo_conjunto = ConjuntoADT()
        nuevo_conjunto.elementos = self.elementos.copy()
        for elemento in otro_conjunto.elementos:
            if elemento not in nuevo_conjunto.elementos:
                nuevo_conjunto.agregar(elemento)
        return nuevo_conjunto
    def interseccion(self, otro_conjunto):
```

PROBLEMS 64 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\pc\Documents\Tareas>
                              > & C:/Users/pc/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/pc/Documents/Tareas/Tarea 2.py"
Conjunto 1: {5, 8, 12, 15}
Conjunto 2: {15, 8, 20}
Unión: {5, 8, 12, 15, 20}
Intersección: {8, 15}
¿Son iguales? False
Eliminando el elemento 12 del conjunto 1: {5, 8, 15}
¿El conjunto 1 contiene al elemento 15? True
¿Conjunto 1 es subconjunto de Conjunto 2? False
Diferencia (Conjunto 1 - Conjunto 2): {5}
PS C:\Users\pc\Documents\Tareas>
```


Tarea 2.py > ...

```python
class ConjuntoADT:
    def __init__(self):
        self.elementos = []

    def agregar(self, elemento):
        if elemento not in self.elementos:
            self.elementos.append(elemento)

    def eliminar(self, elemento):
        if elemento in self.elementos:
            self.elementos.remove(elemento)

    def contiene(self, elemento):
        return elemento in self.elementos

    def union(self, otro_conjunto):
        nuevo_conjunto = ConjuntoADT()
        nuevo_conjunto.elementos = self.elementos.copy()
        for elemento in otro_conjunto.elementos:
            if elemento not in nuevo_conjunto.elementos:
                nuevo_conjunto.agregar(elemento)
        return nuevo_conjunto
    def interseccion(self, otro_conjunto):
        nuevo_conjunto = ConjuntoADT()
        for elemento in self.elementos:
            if otro_conjunto.contiene(elemento):
                nuevo_conjunto.agregar(elemento)
        return nuevo_conjunto

    def __str__(self):
        return f"{{{', '.join(map(str, self.elementos))}}}"

    def longitud(self):
        return len(self.elementos)

    def equals(self, otro_conjunto):
        if self.longitud() != otro_conjunto.longitud():
            return False
        for elemento in self.elementos:
            if not otro_conjunto.contiene(elemento):
                return False
        return True

    def subconjunto(self, otro_conjunto):
        for elemento in self.elementos:
```

Tarea 2.py > ...

```python
class ConjuntoADT:
    def equals(self, otro_conjunto):
        return True

    def subconjunto(self, otro_conjunto):
        for elemento in self.elementos:
            if not otro_conjunto.contiene(elemento):
                return False
        return True

    def diferencia(self, otro_conjunto):
        nuevo_conjunto = ConjuntoADT()
        for elemento in self.elementos:
            if not otro_conjunto.contiene(elemento):
                nuevo_conjunto.agregar(elemento)
        return nuevo_conjunto

conjunto1 = ConjuntoADT()
conjunto1.agregar(5)
conjunto1.agregar(8)
conjunto1.agregar(12)
conjunto1.agregar(15)

conjunto2 = ConjuntoADT()
conjunto2.agregar(15)
conjunto2.agregar(8)
conjunto2.agregar(20)

union = conjunto1.union(conjunto2)
interseccion = conjunto1.interseccion(conjunto2)

print(f"Conjunto 1: {conjunto1}")
print(f"Conjunto 2: {conjunto2}")
print(f"Unión: {union}")
print(f"Intersección: {interseccion}")
print(f"¿Son iguales? {conjunto1.equals(conjunto2)}")

conjunto1.eliminar(12)
contiene = conjunto1.contiene(15)
diferencia = conjunto1.diferencia(conjunto2)

print(f"Eliminando el elemento 12 del conjunto 1: {conjunto1}")
print(f"¿El conjunto 1 contiene al elemento 15? {contiene}")
print(f"¿Conjunto 1 es subconjunto de Conjunto 2? {conjunto1.subconjunto(conjunto2)}")
```