**1.[50 pts] Diligencie la tabla del TAD con todos sus campos: [1pto] nombre, [10 pts] objeto abstracto, [15 pts] invariante, [14 pts] operaciones principales (mínimo 5 además de la constructora), estableciendo entradas, salidas y el tipo de operación (constructora, modificadora o analizadora) y [10 pts] especificación detallada de dichas operaciones. No es necesario definir las precondiciones y postcondiciones de las operaciones.**

|  |
| --- |
| **TAD** 〈Conjunto〉 |
| set “s” = < “s” = <e1, e2, …, en>, size = n>  Conjunto<Integer> a = new Conjunto<>(1, 2, 3, 4);  Conjunto<Integer> b = new Conjunto<>(2, 3, 4, 5); |
| { inv: set.size ≥ 0 ∧ (∀ei∈ set “s” ∧ ∀ej∈ set “s” ∧ i ≠ j ⇒ ei ≠ ej)} |
| **Operaciones Primitivas:**  〈pertenece〉: *Element* → Boolean  〈insertar〉: *Element* → Set  〈Eliminar〉: *Element* → Set  〈Union〉: *Set a Set* → Set  〈Interseccion〉: *Set a Set* → Set |

|  |
| --- |
| pertenece (); |
| Es el conjunto obtenido eliminando el elemento x del conjunto U. |
| public boolean Pertenece(int n) {    System.out.println("s: " + size);  For (int i = 0; i < size; i++) {  if(elements.get(i) == n) {  return true;  }  }  return false;  } |

|  |
| --- |
| Insertar(); |
| Este método se encarga de insertar un elemento n en el conjunto. |
| public void addElement(int n) {    if(!Pertenece(n)) {    elements.add(n);  size++;  }  } |

|  |
| --- |
| Eliminar (); |
| Este método se encarga de eliminar un elemento dentro de un conjunto determinado. |
| public void Eliminar(int n) {  if(Pertenece(n)) {  elements.remove(n);  size--;  }  } |

|  |
| --- |
| Union (); |
| Este método se encargar de realizar la unión de los objetos de los conjuntos a y b. |
| public Set<C1> union(Set<C1> a, Set<C1> b) {  Set<C1> union = new HashSet<C1>(a);  union.addAll(b);  return union;  } |

|  |
| --- |
| Interseccion(); |
| Este método se encarga de intersecar los elementos del conjunto a y b. |
| public Set<C1> intersection(Set<C1> a, Set <C1> b) {  Set<C1> intersection = new HashSet<C1>(a);  intersection.retainAll(b);  return intersection;  } |

**2. Utilizando Generics defina la interfaz genérica e implemente la clase genérica Conjunto a partir de su definición de TAD del punto anterior:**