**TAD de conjunto (Set)**

**Simón García (A00371828)**

|  |
| --- |
| **TAD Set** |
| Set = {Elements = <a1, a2, … an>} para todo a que pertenezca al mismo tipo de dato. |
| {inv: ax != ay tal que x y y son índices en el conjunto y también x != y} |
| Operaciones Primitivas:   * createSet:  -> Set * add: a1 -> Set * delete: a1 -> Set * union: set -> set * intersection: set -> set * difference: set -> set * complement: set -> set * getAllElements: -> List * findElement: a1 -> int * isSubset: set -> boolean * toString: -> String |

|  |
| --- |
| **createSet() - Constructora** |
| “Crea un nuevo conjunto vacío de un mismo tipo de elementos.” |
| True |
| Set = {allElements = { }} |

|  |
| --- |
| **add(a) - Modificadora** |
| “Agrega un nuevo elemento dentro del conjunto” |
| a debe pertenecer al tipo del conjunto y ese elemento no puede repetirse dentro del conjunto |
| Set = {allElements = {a}}, si hay un duplicado no se agrega a los elementos del Set |

|  |
| --- |
| **delete(a) - Modificadora** |
| “Se borra el elemento ingresado del conjunto” |
| a debe ser del mismo tipo del Set y debe hacer parte del conjunto |
| Set = {allElementes = { }}, si no existe, no se cambian los elementos de Set |

|  |
| --- |
| **union(s) - Modificadora** |
| “Se agregan los elementos de otro conjunto” |
| s (s = {a, b}) debe ser un conjunto del mismo tipo que el primer conjunto (Set = {a}) |
| Set = {allElements = {a,b}}, los elementos no se deben repetir. |

|  |
| --- |
| **intersection(s) - Modificadora** |
| “Se cambian todos los elementos dentro del Set por todos los que coincidan con el conjunto secundario.” |
| s (s = {a, x, z}) debe ser del mismo tipo de Set (Set = {a,b,c}) |
| Set = {allElements = {a}} |

|  |
| --- |
| **difference(s, i) - Modificadora** |
| “Se quitan todos los elementos que estén dentro del Set primario que hagan parte del conjunto secundario” |
| s (s = {a, x, z}) debe ser del mismo tipo de Set (Set = {a, b ,c}) y i debe ser 1 o 2. |
| Si i = 1: Set es el primario y se da que Set = {allElementes = {b, c}},  Si i = 2: Set es el secundario y se da que Set = {allElements = {x, z}} |

|  |
| --- |
| **complement(s) - Modificadora** |
| “Se van a cambiar todos los elementos del Set por todos los elementos adicionales distintos que estén en el conjunto secundario” |
| s (s = {a, b, c, x, z}) debe ser del mismo tipo de Set (Set = {a, b ,c}) |
| Set = {allElements = {x , z}}, si Set no es subset de s no se cambian los elementos de Set |

|  |
| --- |
| **getAllElements() - Analizadora** |
| “Se extraen todos los elementos del Set en forma de lista” |
| Set (Set = {a, b ,c}) debe ser un conjunto |
| List<T> = {a, b ,c} |

|  |
| --- |
| **findElement(a) - Analizadora** |
| “Se devuelve la posición en la que se encuentre un elemento” |
| a es un elemento del tipo de Set |
| Devuelve un entero si se encuentra dentro de la lista con la posición o -1 si no existe. |

|  |
| --- |
| **isSubset(s) - Analizadora** |
| “Determina si el Set es un subconjunto del conjunto secundario”. |
| s(s = {a, b, c, x, z}) es del mismo tipo de Set(Set = {a, b ,c}) |
| True si s contiene todos los elementos de Set / False de lo contrario |

|  |
| --- |
| **toString() - Analizadora** |
| “Devuelve una representación amigable del Set en forma de texto”. |
| Set es un Set = {allElements = {a, b, c}} |
| Se devuelven todos los elementos en forma de conjunto de texto: “{a, b, c}” |