

5. Especifica el TAD Conjunto finito de elementos de tipo T. Como constructores considerará el conjunto vacío y el que agrega un elemento a un conjunto. Como operaciones: una que chequee si un elemento *e pertenece* a un conjunto *c*, una que chequee si un conjunto *es vacío*, la operación de *unir* un conjunto a otro, *intersectar* un conjunto con otro y obtener la *diferencia*. Estas últimas tres operaciones deberían especificarse como procedimientos que toman dos conjuntos y modifican el primero de ellos.

```
spec Set of T where
```

constructors

```
fun empty_set() ret s: Set of T
{- crea un conjunto vacío -}

{- PRE:  $e \notin s$  -}
proc add(in/out s: Set of T, in e: T)
{- agrega al elemento e al conjunto s -}
```

destroy

```
proc destroy_set(in/out s: Set of T)
{- libera memoria en caso de que sea necesario -}
```

operations

```
fun member(s: Set of T, e: T) ret b: bool
{- devuelve true si el elemento e pertenece al conjunto s -}

fun is_empty_set(s: Set of T) ret b: bool
{- devuelve true si s es un conjunto vacío -}

proc union(in/out s: Set of T, in s0: Set of T)
{- agrega a s todos los elementos de s0 -}

proc inters(in/out s: Set of T, in s0: Set of T)
{- elimina de s todos los elementos que no pertenezcan a s0 -}

proc diff(in/out s: Set of T, in s0: Set of T)
{- elimina de s todos los elementos que pertenecen a s0 -}
```

```
end spec
```