

# **Informe del Proyecto**

Comision nro6

Schwerdt, Matias David  
Fabrega, Juan Ignacio

## **TDA LISTA**

### **Estructura utilizada para representar la lista**

Se utiliza una lista con enlace simple con un nodo centinela al principio. Se utiliza el concepto de posición indirecta.

### **Operaciones del TDA**

crear\_lista(tLista \* l): Inicializa una lista vacía. Se guarda espacio en memoria, y luego se le asigna nulo a los campos de celda.

l\_insertar(tLista l, tPosicion p, tElemento e): Inserta el elemento e, en la posición p en la lista l. Se chequea que l y p no sean nulos, se guarda espacio en memoria para la nueva posición. Se inserta la nueva posición con rótulo e. Al campo siguiente de la nueva posición se le asigna el siguiente de la posición p. Se setea como siguiente de p, a la nueva posición creada.

l\_eliminar(tLista l, tPosicion p, void (\*fEliminar)(tElemento)): Elimina la celda p de la lista l. El elemento almacenado en la posición p es eliminado mediante la función que es pasada por parámetro. Se controla que p sea una posición correcta. Luego es eliminado el elemento con la función, y se libera el espacio en memoria de la posición eliminada.

l\_destruir(tLista \* l, void (\*fEliminar)(tElemento)): Destruye la lista l, eliminando todas las celdas, los elementos en las celdas son eliminados mediante la función parametrizada.

l\_recuperar(tLista l, tPosicion p): Recupera y retorna el elemento en la posición p. Si p es una posición correcta, se retorna el elemento.

l\_primera(tLista l): Recupera y retorna la primera posición de l. Se retorna l.

l\_siguiente(tLista l, tPosicion p): Recupera y retorna la posición siguiente a p en la lista l. Si p es una posición correcta, se retorna su siguiente.

l\_anterior(tLista l, tPosicion p): Recupera y retorna la posición anterior a p en la lista l. Si p es una posición correcta, se retorna su anterior.

l\_ultima(tLista l): Recupera y retorna la última posición de la lista l.

`l_fin(tLista l)`: Recupera y retorna la posición fin de la lista `l`.

`l_longitud(tLista l)`: Retorna la longitud de la lista `l`. Se recorre la lista mientras el campo siguiente de la posición primera sea distinto de nulo, en cada iteración se actualiza el contador y se avanza la posición.

### Operaciones auxiliares

`destruir_recursivo(tPosicion pos, void (*fEliminar)(tElemento))`

### **TDA ARBOL**