

## ft\_lstsize

La función **ft\_lstsize**, recibe un puntero a una estructura de tipo **t\_list** y devuelve un entero que representa el tamaño de la lista enlazada.

La función inicializa una variable entera **nbr** en 0 y luego itera sobre la lista enlazada hasta que llega al final de la misma, que se indica con un puntero nulo (**void \***)0. En cada iteración, el puntero a la lista se avanza al siguiente nodo con **lst = lst->next;** y se incrementa el contador **nbr++**. Finalmente, se devuelve el valor de **nbr**, que representa el tamaño de la lista enlazada.

Otras partes del código relacionadas con esta función pueden incluir la definición de la estructura **t\_list** y otras funciones que operan sobre esta estructura, como agregar o eliminar nodos de la lista.

### Ejemplo de funcionamiento

Supongamos que tenemos una lista enlazada con los siguientes valores:

1 -> 2 -> 3 -> 4 -> 5 -> NULL

Llamamos a la función **ft\_lstsize** con un puntero a la cabeza de la lista:

```
t_list *lst = malloc(sizeof(t_list));
lst->data = 1;
lst->next = malloc(sizeof(t_list));
lst->next->data = 2;
lst->next->next = malloc(sizeof(t_list));
lst->next->next->data = 3;
lst->next->next->next = malloc(sizeof(t_list));
lst->next->next->next->data = 4;
lst->next->next->next->next = malloc(sizeof(t_list));
lst->next->next->next->next->data = 5;
lst->next->next->next->next->next = NULL;
```

```
int size = ft_lstsize(lst);
printf("La longitud de la lista es: %d\n", size);
```

La salida será:

La longitud de la lista es: 5

Una lista enlazada es una estructura de datos lineal que consiste en un conjunto de nodos, cada uno de los cuales contiene un valor y un puntero al siguiente nodo en la lista. La lista enlazada se puede recorrer iterativamente a través de los punteros, lo que la hace útil para implementar estructuras de datos dinámicas que pueden crecer o encogerse en tiempo de ejecución.

Las listas enlazadas se utilizan en una variedad de aplicaciones, como la gestión de memoria, la simulación de sistemas complejos y la representación de datos en árboles y grafos. Algunos ejemplos prácticos de su uso incluyen:

- Implementación de una cola de procesos en un sistema operativo.
- Almacenamiento de datos en una base de datos relacional.
- Representación de una red social en un grafo.
- Implementación de una pila de operaciones matemáticas.
- Gestión de memoria en un compilador o intérprete de lenguaje de programación.