

Unidad 4: Listas

Autor: PABLO CUESTA SIERRA

Grupo: 210

Listas enlazadas

1. Implementar con control de errores una función no recursiva en C llamada `list_tam` que devuelva el número de nodos de la lista enlazada proporcionada. Suponga las siguientes estructuras y tipos en C:

```
// En list.h
typedef struct _List List;

// En list.c
typedef struct _Node {
    Element *info;
    struct _Node *next;
};
typedef struct _Node Node;
struct _List {
    Node *first;
};
```

- Función implementada sin recursividad (suponemos que podemos acceder a la estructura de lista y de nodo):

```
long list_tam(List *pl){
    long count;
    Node *pn = NULL;

    if(pl == NULL) return -1;

    if(list_isEmpty(pl) == TRUE) return 0;

    for(pn = pl->first, count = 1; pn->next != NULL; pn = pn->next, count++);

    return count;
}
```

2. Implementar recursivamente la función anterior.

```
long list_tam(List *pl){
    long count;

    if(pl == NULL) return -1;
    if(list_isEmpty(pl) == TRUE) return 0;

    /* llamada a la función recursiva */
    count = list_tam_rec(pl->first);

    return count;
}

/*función recursiva */
long list_tam_rec(Node *pn) {
    long count = 1;

    if(pn->next == NULL)
        return count;

    count += list_tam_rec(pl->next);

    return count;
}
```