

1 Jalon 1 : module Liste

Le module liste gère une liste simplement chaînée. Les données enregistrées dans les noeuds sont des pointeurs génériques (`void *`). La gestion mémoire des objets pointés n'est prise en charge dans le module. Une liste vide sera représentée par un pointeur `NULL`.

Le module devra apporté les fonctionnalités minimales suivantes :

- création d'une liste,
- insertion d'une donnée en tête de liste,
- obtention du noeud suivant,
- ajout d'une donnée en queue de liste,
- lecture et écriture de la données associée à un noeud,
- suppression d'une donnée en tête de la liste,
- suppression de la première instance d'une donnée dans la liste,
- destruction d'une liste complète.

L'interface publique du module est donnée ci-dessous.

```
typedef struct node node_t;
node_t * list_create(void);
    // creation d'une nouvelle liste vide
void * list_get_data(const node_t * node);
void list_set_data(node_t * node, void * data);
    // lire ou ecrire la donnee d'un noeud
node_t * list_next(node_t * node);
    // obtenir le noeud suivant
node_t * list_insert(node_t * head, void * data);
    // creation et insertion d'un noeud en tete de liste
    // retourne la tete de liste
node_t * list_append(node_t * head, void * data);
    // creation et ajout d'un noeud en queue de liste
    // retourne la tete de liste
node_t * list_remove(node_t * head, void * data);
    // suppression de la premiere instance d'une
    // donnee dans la liste, retourne la tete de liste
node_t * list_headRemove(node_t * head);
    // suppression de la tete de liste
    // retourne la nouvelle tete de liste
void list_destroy(node_t * head);
    // destruction d'une liste
    // (La libération des données n'est pas prise en charge)
```