



CPPLI : TD 5 : C : Liste circulaire simplement chaînée

Romain Absil Jonas Beleho Pierre Bettens
David Hauweele Pierre Hauweele
Nicolas Vansteenkiste * (ESI – HE2B)

Année académique 2021 – 2022

Ce TD¹ aborde l'implémentation en langage C du type abstrait de données appelé *liste circulaire simplement chaînée*².

Ex. 5.1 Soit :

```
struct SLNode
{
    struct SLNode * next;
    value_t        value;
};
```

Il s'agit d'un type structuré qui sert à représenter les éléments d'une *liste simplement chaînée*³. La signification précise de ses champs est décrite dans le fichier `slnode.h`⁴.

*Et aussi, lors des années passées : Monica Bastreggi, Stéphan Monbaliu, Anne Rousseau et Moussa Wahid.

1. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/td05_c.pdf (consulté le 13 septembre 2021).

2. https://en.wikipedia.org/wiki/Linked_list#Circular_linked_list (consulté le 13 septembre 2021).

3. https://fr.wikipedia.org/wiki/Liste_cha%C3%A9e#Liste_simplement_cha%C3%A9e (consulté le 13 septembre 2021).

4. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/ressource/slnode.h (consulté le 13 septembre 2021).

Le type `value_t` est un [alias de type](#)⁵ quelconque. Il est défini dans le fichier `value_t.h`⁶.

Implémentez les fonctions de manipulation de `struct SLNode` suivantes, dont une [documentation](#)⁷ précise est fournie comme pour [doxygen](#)⁸ dans `slnode.h` :

```
struct SLNode * newSLN(value_t value);

void deleteSLN(struct SLNode * * adpSLN);

struct SLNode * nextSLN(const struct SLNode * pSLN);

void setNextSLN(struct SLNode * pSLN, struct SLNode * pNewNext);

value_t valueSLN(const struct SLNode * pSLN);

void setValueSLN(struct SLNode * pSLN, value_t newValue);
```

Ex. 5.2 Implémentez la fonction suivante facilitant l'utilisation de `struct SLNode` (voir Ex. 5.1), dont une documentation précise est fournie dans `slnode_utility.h`⁹ :

```
struct SLNode * forwardSLN(struct SLNode * pSLN, size_t distance);
```

Ex. 5.3 Soit :

```
struct SLCircularList
{
    struct SLNode * entry;
};
```

Le type structuré `struct SLCircularList` sert à représenter une liste circulaire simplement chaînée. Le type de son champ `entry` est `struct SLNode *` tel que défini à l'Ex. 5.1. La signification précise de ce champ est décrite dans le fichier `slcircularlist.h`¹⁰.

Implémentez les fonctions de manipulation de `struct SLCircularList` suivantes, dont une documentation précise est fournie dans `slcircularlist.h` :

5. <https://en.wikipedia.org/wiki/Typedef> (consulté le 13 septembre 2021).

6. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/ressource/value_t.h (consulté le 13 septembre 2021).

7. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/ressource/html.tar.xz (consulté le 13 septembre 2021).

8. <http://www.doxygen.nl/> (consulté le 13 septembre 2021).

9. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/ressource/slnode_utility.h (consulté le 13 septembre 2021).

10. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/ressource/slcircularlist.h (consulté le 13 septembre 2021).

```
struct SLCircularList * newSLCL(void);

void deleteSLCL(struct SLCircularList * * adpSLCL);

void clearSLCL(struct SLCircularList * pSLCL);

struct SLNode * entrySLCL(const struct SLCircularList * pSLCL);

bool emptySLCL(const struct SLCircularList * pSLCL);

struct SLNode * pushSLCL(struct SLCircularList * pSLCL,
                        value_t value);

struct SLNode * insertSLCL(struct SLCircularList * pSLCL,
                          struct SLNode * pSLN,
                          value_t value);

struct SLNode * popSLCL(struct SLCircularList * pSLCL);

struct SLNode * eraseSLCL(struct SLCircularList * pSLCL,
                        struct SLNode * pSLN);
```

Ex. 5.4 Implémentez les fonctions d'utilisation de `struct SLCircularList` (voir Ex. 5.3) suivantes, dont une documentation précise est fournie dans le fichier d'en-têtes `slcircularlist_utility.h`¹¹ :

```
size_t sizeSLCL(const struct SLCircularList * pSLCL);

struct SLNode * previousSLCL(const struct SLCircularList * pSLCL,
                          const struct SLNode * pSLN);
```

11. https://poesi.esi-bru.be/pluginfile.php/7311/mod_folder/content/0/td05_c/ressource/slcircularlist_utility.h (consulté le 13 septembre 2021).