

Ressource disponible : module pile

Le module pile prend en charge une structure de données de type pile LIFO. Le nombre maximum d'éléments est fixé au moment de la création de la pile.

Ce module est déjà implanté, et ne doit pas être ré-écrit. Son code est disponible sur page Moddle du projet.

Le module apporte les fonctionnalités suivantes :

- création et destruction d'une pile,
- empilement et dépilement d'une valeur,
- nombre de places restantes et occupées,
- lecture des valeurs dans la pile.

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

```
typedef struct my_stack my_stack_t;
my_stack_t * stack_create(int size_max);
    // retourne NULL en cas d'erreur
void stack_remove(my_stack_t * p);
#define STACK_CAPACITY(stack)
    // retourne le nombre maximum d'éléments dans la pile
#define STACK_EMPTY(stack)
    // retourne 1 si la pile est vide, 0 sinon
#define STACK_MEM_AVAILABLE(stack)
    // retourne le nombre de places restant libres dans la pile
#define STACK_MEM_USED(stack)
    // retourne le nombre de places utilisées dans la pile
#define STACK_POP(stack, type)
    // retourne la valeur en sommet de pile, 0 si la pile est vide
#define STACK_POP2(stack, var, type)
    // affecte la valeur dépiler à var
    // retourne -1 si la pile est vide, 0 sinon
#define STACK_PUSH(stack, val, type)
    // empile la valeur de val
    // retourne -1 si la pile est pleine, 0 sinon
#define STACK_INIT_ITERATION(stack, var, type)
    // initialise l'itérateur
#define STACK_ITERATE(stack, pos, type)
    // affecte la valeur courrante à val
    // retourne 1 si une nouvelle valeur est disponible, 0 sinon
```