



Ensembles et allocation dynamique

En cours on a vu un ensemble de fonctions qui permettent de faire de l'allocation dynamique :

```
void *malloc(size_t size);  
void free(void *ptr);  
void *calloc(size_t nmemb, size_t size);  
void *realloc(void *ptr, size_t size);
```

Dans ce TD, on va apprendre à les utiliser et définir une structure qui permet de stocker un ensemble d'entiers dont le cardinal n'est pas fixé à l'avance. On remarque que cette notion d'ensemble peut être étendue à n'importe quel type de donnée. En cours, nous avons vu la structure `garage`, qui est un ensemble de voitures.

► Exercice 1. Structure ensemble d'entiers

- Définir une structure `ensemble_t` pouvant contenir des entiers. On aura besoin de 3 champs :
 1. un tableau pour stocker les entiers
 2. un entier indiquant le nombre de cases du tableau
 3. un entier indiquant le cardinal de l'ensemble.
- Écrire la fonction `ensemble_initialiser` qui crée un ensemble vide. On allouera deux espaces mémoires : la structure en elle-même et un tableau de 10 cases pour stocker les entiers.
- Écrire la fonction `ensemble_vide` qui renvoie 1 si l'ensemble est vide et 0 sinon.
- Écrire la fonction `ensemble_cardinal` qui renvoie le nombre d'entiers stockés dans l'ensemble.
- Écrire la fonction `ensemble_appartient` qui prend en entrée un `ensemble_t e` et un entier x , renvoie 1 si $x \in e$ et 0 sinon.
- Écrire la fonction `ensemble_ajouter` qui prends en entrée un pointeur sur un `ensemble_t` et un entier x et ajoute x à l'ensemble. Note : si le tableaux utilisé pour stocker les éléments est plein, on double sa taille à l'aide de la fonction `realloc`. De plus, il faut s'assurer que x n'appartient pas déjà au tableau.
- Écrire la fonction `ensemble_supprimer` qui prends en entrée un pointeur sur un `ensemble_t` et un entier x et supprime x de l'ensemble.
- Écrire la fonction `ensemble_détruire` qui prends en entrée un pointeur sur un `ensemble_t` et désalloue l'espace mémoire.
- Écrire la fonction `ensemble_afficher` qui prend en entrée un `ensemble_t` et affiche son contenu.
- Écrire la fonction `ensemble_union` qui prends en entrée deux structures `ensemble_t X` et Y et renvoie un troisième ensemble $Z = X \cup Y$.
- Écrire la fonction `ensemble_intersection` qui prends en entrée deux structures `ensemble_t X` et Y et renvoie un troisième ensemble $Z = X \cap Y$.
- Quelle est la complexité en temps et en espace de toutes ces fonctions ?