

PARTIEL 1

Durée : 1h30

Seuls sont autorisés, à titre de documents, les listings comprenant explicitement le nom de l'étudiant imprimé (et celui-ci seulement) et les notes manuscrites (pas de photocopie) comprenant le nom de l'étudiant sur chaque page.

Il est conseillé d'écrire les exercices en langage C++. L'écriture en Java complique les entrées-sorties et l'écriture en langage C allonge le programme (ce qui ne rapportera pas de point supplémentaire dans un cas comme dans l'autre).

Exercice unique.- Multi-ensemble d'entiers

*Un ensemble contient des éléments, une seule fois chaque élément : l'ensemble $\{a, b, a\}$ est égal à l'ensemble $\{a, b\}$. Il existe certaines situations où on voudrait généraliser la notion d'ensemble pouvant contenir plusieurs fois le même élément, c'est-à-dire où $\{a, b, a\}$ serait différent de $\{a, b\}$. On parle alors de **multi-ensemble** (multiset ou bag en anglais).*

- 1^o) Implémenter une classe **Bag** de chaînes de caractères comprenant :

- un constructeur par défaut (multi-ensemble vide ne pouvant contenir au plus qu'un élément),
- un constructeur avec un argument entier naturel spécifiant le nombre maximum d'éléments du multi-ensemble, vide au départ,
- une méthode `full()` qui indique si le multi-ensemble est plein (c'est-à-dire s'il contient le nombre maximum d'éléments permis),
- une méthode `empty()` qui indique si le multi-ensemble est vide,
- une méthode `add()` dont le seul paramètre est l'entier dont une occurrence est à ajouter au multi-ensemble (si celui-ci n'est pas plein),
- une méthode `remove()` dont le seul paramètre est l'élément dont une occurrence est à retirer (s'il appartient au multi-ensemble) et qui renvoie 1 si on a retiré une occurrence, 0 sinon,
- une méthode `occurrences()` dont le paramètre est un entier et qui renvoie le nombre d'occurrences de cet entier dans le multi-ensemble,
- une méthode `size()` qui renvoie le nombre (d'occurrences) d'éléments du multi-ensemble.

[*Un multi-ensemble d'entiers sera implémenté comme un tableau d'entiers (partiellement rempli). Un élément ajouté sera placé à la fin du tableau (partiellement rempli). Retirer un élément consistera à enlever une occurrence de l'élément et à la remplacer par le dernier élément du tableau.*]

- 2°) Écrire un programme de test demandant les âges des membres de votre famille (de moins de vingt personnes) en terminant par un entier négatif. Pour vérification, on redemande les âges, que l'on peut donner dans un autre ordre mais toujours en terminant par un entier négatif. Le programme doit tester la cohérence des données.

[*Un exemple de session est :*

```
$ ./a.out
Age : 10
Age : 15
Age : 13
Age : 15
Age : -1
Retapez l'age des membres de votre famille :
Age : 13
OK
Age : 9
Cet age n'apparait pas !
Age : 15
OK
Age : 15
OK
Age : -1
Vous en avez oublie
$
```

]