

## A113 – TD2

### Manipulation de tableaux

#### Exercice 1

Écrire les fonctions ou les procédures suivantes :

- ✓ `estTrieCroissant` qui détermine si un tableau de réels de taille quelconque est trié dans l'ordre croissant ;
- ✓ `estTrieDeroissant` qui détermine si un tableau de réels de taille quelconque est trié dans l'ordre décroissant ;
- ✓ recherche permettant de connaître la position d'une valeur dans un tableau de réels, -1 si non trouvé ;
- ✓ `rechercheDico` permettant de faire une recherche dichotomique d'une valeur dans un tableau de réels trié en ordre croissant ;
- ✓ `triSelection` permettant de trier un tableau de réels à l'aide d'un tri par sélection ;
- ✓ `triBulle` permettant de trier un tableau de réels à l'aide d'un tri par bulle ;
- ✓ `suppElemPos` permettant de supprimer un élément dans un tableau de réels à une position donnée ;
- ✓ `suppElem` permettant de supprimer un élément à partir de sa valeur dans un tableau de réels (premier élément trouvé), utiliser la fonction recherche et `suppElemPos` précédentes ;
- ✓ `suppTous` permettant de supprimer toutes les occurrences d'un élément dans un tableau de réels ;
- ✓ `InserElemPos` permettant d'insérer un élément dans un tableau de réels à une position donnée ;
- ✓ `InserElem` permettant d'insérer un élément dans un tableau de réels trié.
- ✓ Écrire un programme utilisant les fonctions précédentes.

#### Exercice 2

Afficher la durée de traitement de chaque fonction. Que pouvez-vous en conclure, notamment pour les 2 sous-programmes de tris ?

Pour cela utiliser la fonction suivante pour initialiser un très grand tableau (n=10000) de manière aléatoire.

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
void init(float* unTab, int unN)
{
    int i;
    srand(time(NULL));
    for (i = 0; i < unN; i++)
        unTab[i] = (float)rand();
}
```

Pour afficher la durée d'exécution d'un traitement, encadrer votre traitement avec les instructions suivantes :

```
int debut,fin;
int n=10000 ;
float tab[10000];
debut= clock()
init(tab,n);
fin= clock();
printf("  durée  de  l'initialisation%f\n", (double)(fin-debut) / (double)
CLOCKS_PER_SEC);
```