

## ESIEA

### ALGORITHMIQUE ET PROGRAMMATION-INF2033

#### TD2

#### EXERCICE 1

- 1) Ecrivez dans le langage C une **procédure** qui prend **en entrée l'horaire de départ d'un vol et l'horaire de retour du même vol et retourne la durée du vol**. On supposera que la durée d'un vol ne dépasse pas une journée. Le prototype de cette procédure est le suivant

```
void dureevol(int hd, int md, int sd, int *ha, int *ma, int *sa, int *h, int *m, int *s);
```

- 2) Complétez la fonction main.

#### EXERCICE 2

Soit  $T[1 \dots n]$  un tableau de  $n$  réels.

- 1) Ecrivez une **procédure** qui prend **en entrée le tableau T et retourne TP, TN, nbp et nbn** où,
- ✓ TP est le tableau extrait de T et qui ne contient que les réels positifs ou nuls ;
  - ✓ nbp est le nombre d'éléments dans TP ;
  - ✓ TN est le tableau extrait de T et qui ne contient que les réels négatifs ;
  - ✓ nbn est le nombre d'éléments dans TN.

#### EXERCICE 3

Soit T un tableau dynamique de dimension 1.

- 1) Ecrivez dans le langage C une **procédure** qui permet de remplir aléatoirement le tableau T par des entiers compris entre 1 et 99 et qui retourne l'adresse du tableau. Le prototype de cette procédure est le suivant

```
int *chargement_aleatoire(int *tab, int nb);
```

- 2) Ecrivez dans le langage C une **procédure** qui prend en entrée le tableau T et qui

retourne le maximum de T ainsi que sa position. Le prototype de cette procédure est le suivant

```
void tabmax_place(int T[], int n, int *max, int *place) ;
```

3) Complétez la fonction main.

#### **EXERCICE 4**

Soit **T[1 n]** un tableau **non trié** de n entiers. On désire écrire un algorithme permettant de trier le tableau T[1 n] dans l'ordre croissant.

Pour ce faire, on procède comme suit. **Pour i allant de n jusqu'à 2, on parcourt le tableau T[1 i] de gauche à droite et, chaque fois que deux éléments consécutifs ne sont pas dans le bon ordre, on les permute.**

1) Ecrivez dans le langage C la **procédure itérative** correspondante. Le prototype de cette procédure est le suivant

```
int *bulle(int T[], int n);
```

Soit **T[1 n]** un tableau **trié** de n entiers. On souhaite ajouter un élément X au tableau T de sorte que le tableau résultant soit encore trié. Le schéma de l'algorithme est le suivant.

Procédure insertion (dr T : tableau [1 ... n] de type t ; dr n : entier ; d X : t) ;

{...}

Var ind : entier

Début

Si n = 0 alors

n := 1

T[1]:=X

Sinon

ind := place (T, n, X)

insertind (T, n, X, ind)

Fin ;

A cette fin, on utilisera les prototypes suivants

```
void affiche(int *tab, int nb);
```

```
int *chargement_aleatoire(int *tab, int nb);
```

```
int *bulle(int T[], int n);
```

```
int place(int T[], int n, int X);  
void insertind(int T[], int *n, int X, int ind);
```

où,

affiche est une procédure qui affiche les éléments du tableau ;

place est une fonction qui donne la place "ind" de l'élément à insérer X ;

insertind est une procédure qui effectue l'insertion de l'élément X à la place "ind".

- 2) Ecrivez dans le langage C les **prototypes** qui précèdent.
- 3) Complétez la fonction main et ce, en partant d'un tableau que l'on remplira aléatoirement par des entiers compris entre 1 et 99.