

## TD9

Tous les exercices de ce TD doivent utiliser la notation pointeur pour parcourir les tableaux.

**Exercice 1 :** écrivez une procédure qui sépare un tableau de 20 entiers en 2 tableaux : un tableau ne contenant que les nombres pairs et un tableau ne contenant que des nombres impairs.

**Exercice 2 :** Écrivez la structure de données Point2D. Un point doit avoir un nom en plus de ses coordonnées. Pour plus de facilité dans les affichages, les coordonnées du point sont en int.

**Exercice 3 :** Écrivez la structure de données Rectangle. Un rectangle doit également avoir un nom. Les valeurs qui caractérisent un rectangle seront déclarées en entiers pour faciliter l’affichage.

**Exercice 4 :** Écrivez la procédure consPoint qui permet de construire un point.

**Exercice 5 :** Écrivez la procédure consRectangle qui permet de construire un rectangle.

**Exercice 6 :** Écrivez une procédure affichePoint qui affiche de manière textuelle un Point2D (cf affichage dans la copie d’écran de l’exercice 10).

**Exercice 7 :** Écrivez une procédure afficheRectangle qui affiche un rectangle. Nous pourrons réutiliser le programme qui avait été écrit pour dessiner un rectangle avec des étoiles.

**Exercice 8 :** la procédure modifPoint qui modifie un Point2D.

Vous passerez en paramètre de cette procédure un pointeur sur un Point2D ainsi que les nouvelles valeurs nécessaires pour mettre à jour ce point.

**Exercice 9 :** Écrivez la procédure `modifRectangle` qui modifie un `Rectangle`.

Vous passerez en paramètre de cette procédure un pointeur sur un `Rectangle` ainsi que les nouvelles valeurs nécessaires pour mettre à jour cette figure.

**Exercice 10 :** Écrivez le main qui permet d'obtenir le résultat suivant :

```

c:\Enseignement\2013_2014\M1102\TD2\Debug\TD2.exe
Programme qui cree, modifie et affiche un rectangle
Pour cela, vous allez d'abord creer un P0int2D ppuis un rectangle

Donnez les valeurs nom, x et y du point
A
2
2
A (2, 2)

Donnez les valeurs nom, hauteur et largeur du rectangle
Rect1
5
15
A :
*****
*****
*****
*****
*****

Affichage du point apres avoir rajoute 2 pour X et Y
A (4, 4)

Affichage du rectangle apres avoir rajoute 4 pour hauteur et largeur
Rect1 :
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
Appuyez sur une touche pour continuer... _

```

**Exercice 11 :** Écrivez la procédure qui permet d'initialiser un tableau de N rectangles. N est passé en paramètre de la procédure. Pour initialiser les valeurs des points et des hauteurs/largeur des rectangles, vous utiliserez la fonction `rand` (cf. exemple d'utilisation ci-dessous).

Pour les coordonnées des points, vous utiliserez des nombres compris entre 0 et 10. Pour la hauteur et la largeur des rectangles, vous utiliserez des nombres compris entre 0 et 20.

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

```
...
```

```
time_t nb_secondes;
```

```
srand (time(&nb_secondes)) ;    // cette instruction ne doit être appelée
qu'une seule fois.
```

```
x = rand ()% 10 ;           // génère un nombre aléatoire compris entre 0 et 10
```

**Exercice 12 :** Écrivez la fonction qui calcule la surface d'un rectangle.

**Exercice 13 :** Écrivez la fonction qui affiche le rectangle qui a la plus grande surface dans le tableau de rectangles initialisé dans l'exercice 10.