

Pointeurs

Exercice 1

Soit la structure suivante

```
typedef struct{
    int x, y;
} point_t;
```

Ecrire une fonction

```
void init_point_1(int v1, int v2, point_t *p);
```

qui initialise les champs d'un point passé par référence ainsi qu'une fonction

```
point_t init_point_2(int v1, int v2);
```

qui initialise un point et le retourne.

Ecrivez un programme principal qui utilise ces deux fonctions;

Exercice 2

Ecrire une fonction `divide` qui prend en paramètres deux nombres entiers `a` et `b` et un pointeur sur un nombre entier `res` et retourne un nombre entier.

La fonction calcule le résultat de la division entière `a/b` et stocke le résultat dans `res`. Si `b!=0`, la fonction retourne 0, sinon elle retourne 1 et la valeur stockée dans `res` est indéfinie.

Exercice 3

Écrire une fonction `swap` qui a comme paramètres deux pointeurs vers des entiers et qui échange le contenu des deux entiers pointés. Ecrire un programme qui échange le contenu de deux entiers `a` et `b` en appelant cette fonction.

Exercice 4

Écrire une fonction qui a comme paramètres un tableau d'entiers de taille quelconque, la taille du tableau, et 2 pointeurs vers des entiers `min` et `max`. La fonction doit écrire dans les entiers pointés par `min` et `max` respectivement les plus petits et les plus grands entiers du tableau.

Exercice 5

Écrire une fonction qui a en paramètre un tableau de nombres entiers et qui renvoie le nombre d'occurrences du nombre 10. Cette fonction devra parcourir le tableau en utilisant un pointeur.