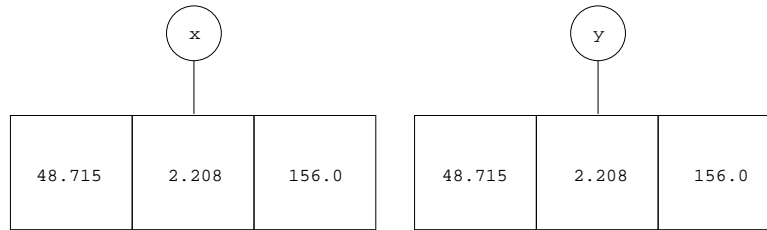


et cet enregistrement est recopié dans la variable `y` au moment de l'appel de la fonction.



Ainsi, le programme

```
int main () {
    struct Point x = {90.0,0.0,0.0};
    x.latitude = 48.715;
    x.longitude = 2.208;
    x.altitude = 156.0;
    tropic(x);
    printf("%f\n",x.latitude);
    return 0;}
```

affiche 48.715 et non 23.45.

L'utilisation d'une structure ne permet donc pas d'écrire des fonctions en appel par référence. Pour qu'une fonction qui prend un enregistrement en argument puisse modifier les champs de cet enregistrement, il faut utiliser la construction `&`, comme dans le cas d'un argument de type scalaire.

```
void tropic (struct Point* y) {
    (*y).latitude = 23.45;}

tropic(&x);
```

L'instruction

```
struct Point x = {90.0,0.0,0.0};
struct Point* y = &x;
```

construit l'état

