

```

// la fonction ft_ft prend en argument
// un pointeur (*) vers un entier (int)
// ce pointeur est appelé nbr

// il pointe vers l'adresse de la variable de type int passée en paramètre
// il contiendra donc cette adresse
// ATTENTION : c'est bien l'adresse de la variable de type int qui doit être
// passée en paramètre, et non pas directement la variable

void    ft_ft(int *nbr)
{
    // ATTENTION : ici, *nbr est une "lvalue"
    // car elle se trouve à gauche (left) de l'opérande
    // d'affectation =
    // * est donc ici l'opérateur de déréférencement
    // cela signifie :
    // variable située à l'adresse fournie par le pointeur nbr
    // cette instruction modifie la valeur de la variable pointée par nbr
    // (la valeur de la variable se trouvant à l'adresse contenue dans *nbr,
    // donc à l'adresse passée en paramètre de la fonction)
    // en lui assignant la valeur 42
    *nbr = 42;
}

// main.c :
#include "ft_ft.h"

// on inclut le fichier d'entête de la fonction externe ft_putnbr
#include "ft_putnbr.h"

int     main (void)
{
    // on définit une variable de type int appelée n
    int    n;

    // on exécute ft_ft en lui passant l'adresse
    // de la variable n de type int
    // & signifie "adresse de "

```

```
ft_ft(&n);

// pour afficher notre nombre :
ft_putnbr(n);

// pour indiquer au système d'exploitation que le programme s'est terminé avec succès
return (0);
}

// ATTENTION :
// POUR QUE CELA FONCTIONNE, IL FAUT COPIER ft_putnbr.c ET ft_putnbr.h
// (de c00/ex07) DANS LE DOSSIER c01/ex00
// PUIS COMPILER LES 3 FICHIERS C
// (ft_ft.c, ft_putnbr.c et main.c)
// EN MEME TEMPS
// avec la commande :
// gcc -Wall -Wextra -Werror ft_ft.c ft_putnbr.c main.c
```