

```

// ce programme (composé d'une fonction main)
// affiche le nom du programme :
// compilé avec gcc -Wall -Wextra -Werror ft_print_program_name.c
// puis exécuté avec ./a.out
// doit donc afficher ./a.out à l'écran

// pour pouvoir utiliser la fonction write
#include <unistd.h>

// le main prend en paramètre le nombre d'arguments,
// l'entier argc
// et un pointeur de pointeur argv
// (de type pointeur sur un pointeur de caractères)
// qui pointe sur le pointeur des arguments

// c'est un tableau de chaînes de caractères
// (chaque chaîne de ce tableau étant un pointeur
// sur le premier caractère de cette chaîne)

// chaque chaîne représentant un argument passé au programme
// le premier argument étant le nom du programme lui-même
// argv[0] sera donc un pointeur sur le premier caractère
// du nom du programme

int    main(int argc, char **argv)
{
    // on déclare un pointeur str
    // qui stockera (voir plus bas)
    // le pointeur vers la chaîne de caractères argv[0]
    char    *str;

    // IMPORTANT :
    // on indique au compilateur qu'argc ne sera pas utilisé
    // SINON, AVERTISSEMENT DU COMPILATEUR POUR VARIABLE DECLARÉE
    // MAIS NON UTILISÉE !
    (void) argc;

    // on affecte à str le premier argument passé au programme

```

```

// qui est le pointeur vers le premier caractère du nom du programme
str = argv[0];

// POUR PARCOURIR CHACUN DES CARACTÈRES POINTÉS PAR str
// EN INCREMENTANT str AU FUR ET À MESURE
// JUSQU'À ATTEINDRE LE CARACTÈRE NUL '\0'
// MARQUANT LA FIN DE LA CHAÎNE
while (*str != '\0')
    // on écrit le caractère en cours
    // pointé par str
    // en l'incrémentant au fur et à mesure
    // write(1, str++, 1); écrira str
    // puis l'incrémentera de 1
    // (identique à :
    // write(1, str, 1);
    // str++;

    // car str++ est une post-incrémentation
    write(1, str++, 1);
// on retourne 0 pour indiquer que le programme s'est terminé avec succès
return (0);
}

```