

paramètres de main()

les paramètres de main()

```
int main( int argc, char *argv[] ){ }
```

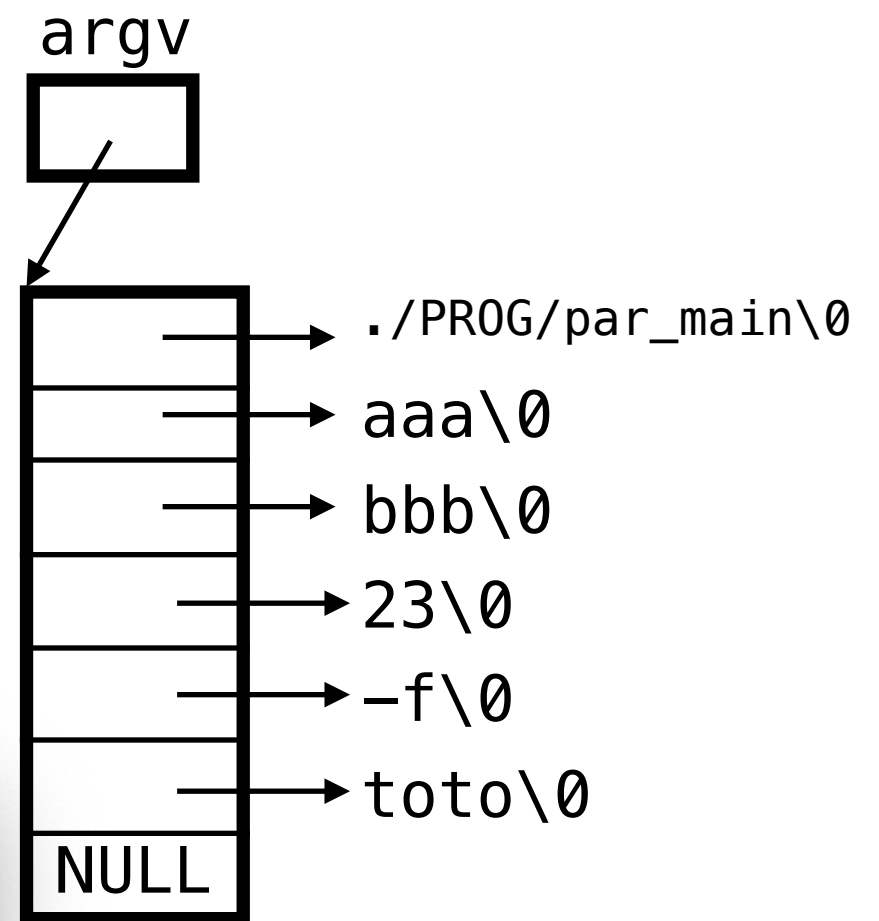
```
int main( int argc, char **argv ){ }
```

argc == 6

Supposons que le programme compilé `par_main` suivant se trouve dans le répertoire `PROG`

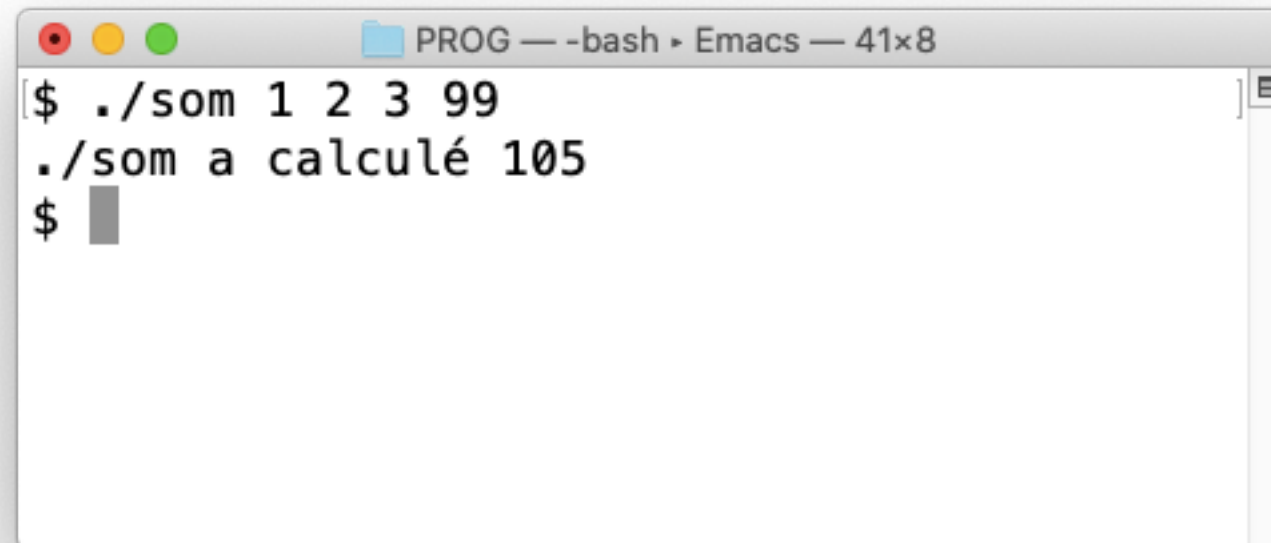
```
/* par_main.c */
#include <stdio.h>
int main(int argc, char *argv[]){
    for(int i = 0; i < argc; i++){
        printf("%s\n", argv[i]);
    }
}
```

```
COURS06 — -bash • Emacs — 49x9
$ ./PROG/par_main aaa bbb 23 -f toto
./PROG/par_main
aaa
bbb
23
-f
toto
$
```



paramètres de main()

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char **argv){
    int s = 0;
    for(int i = 1 ; i < argc; i++){
        s += atoi( argv[i] );
    }
    printf( "%s a calculé %d\n", argv[0] , s);
}
```

A terminal window titled 'PROG — -bash > Emacs — 41x8' showing the execution of a program named 'som'. The user enters './som 1 2 3 99' and the program outputs './som a calculé 105'. The prompt '\$' is followed by a cursor.

```
PROG — -bash > Emacs — 41x8
[$ ./som 1 2 3 99
./som a calculé 105
$ █
```

terminaison de programme

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

```
void exit( int status )
```

La fonction `exit()` termine l'exécution de programme. Le paramètre : le code d'exit.

`exit(0)` – terminaison normale, `exit(i)` avec `i > 0` terminaison avec erreur ,

En pratique les valeurs `exit()` à utiliser entre 0 et 127.

```
int main(int arc , char **argv){  
    int i = ...  
    return i; /*    return i; dans main() le même effet que  
                *    exit(i)                                */  
}
```

(sauf si `main()` est appelé depuis une autre fonction ou récursivement -- oui c'est possible).

Mais on peut aussi quitter **`main()`** sans **`exit()`** ni **`return`**, juste parce qu'on a exécuté la dernière instruction de `main()`, dans ce cas `main()` termine avec la valeur de retour 0;
(Mais c'est valable uniquement pour `int main()` et pour aucune d'autre fonction qui retourne `int`.)

terminaison de programme

Comment voir le code de retour depuis le terminal? Après la terminaison de programme lancé depuis le terminal tapez sur le terminal

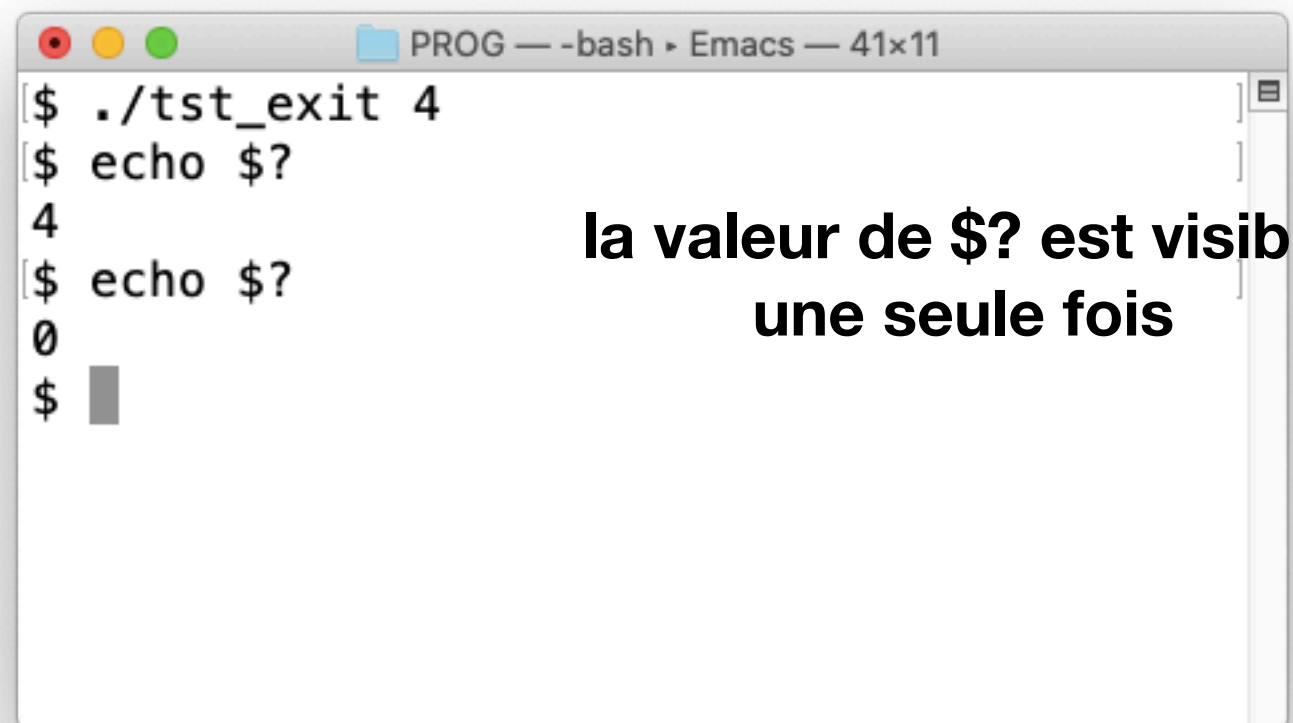
`echo $?`

Cela affiche la valeur de la variable `?` de `shell` : le code de retour de la dernière commande/programme C exécutée sur le terminal.

```
/* tst_exit.c */
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char **argv){
    if( argc != 2 ){
        printf("%s code\n", argv[0] );
        exit(0);
    }

    int a = atoi(argv[1]);
    exit(a);
}
```



```
PROG — -bash • Emacs — 41x11
$ ./tst_exit 4
$ echo $?
4
$ echo $?
0
$
```

**la valeur de `$?` est visible
une seule fois**