

Algo & Prog pour l'ingénieur

TP 1

Les tous premiers pas avec le langage C...

Exercice 1 Premier programme C

Et bien il s'agit bien sûr du Hello World! Quitte à le recopier entièrement, faites le quand même! Compilez-le vous même! Ensuite, faites un Makefile pour n'avoir plus besoin que de taper make dans un terminal pour le compiler.

Exercice 2 Faire une addition

Écrire un programme C qui, lors de son lancement, demande deux entiers à l'utilisateur, qui les tapera alors en console. Le programme calculera alors la somme de ces deux entiers et affichera le résultat dans la console.

Voir les fonctions (et leurs pages de manuel) `scanf`, `printf` et en particulier le `%d` des expressions formatés.

Modifier le programme précédant pour que l'utilisateur donne les deux entiers via la ligne de commande à l'appel du programme. les arguments passeront alors par les variables `argc` et `argv[]` de votre `main`. Pour la conversion des chaînes de caractères (arguments du `main`) en entiers, voir la fonction `atoi`.

Exercice 3 Égrainer des nombres

Écrire une fonction itérative (avec des `for`, `while` ou `do...while`) qui prend en argument un entier n et provoque l'affichage suivant pour $n = 10$.

10 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7 8 9 10

Écrire une fonction récursive (une fonction qui se rappelle elle-même dans son code) qui produit le même effet.

Exercice 4 Pointeur vers un entier

Recopiez le programme suivant et tentez de comprendre ce qui se passe. Attention, tout copier-coller depuis un pdf donnera souvent un code non compilable... Il faut le réécrire! (Pour les curieux, les copier-coller depuis un pdf génèrent des caractères non ASCII).

```
1 #include <stdio.h>
2
3 void print_var(int n){
4     printf("Value of the variable: %d\n", n);
5 }
6
7 void print_pointer(int* p){
8     printf("Pointer address: %p and Pointed value: %d\n", p, *p);
9 }
```

```
10
11 void set_pointer(int* p, int n){
12     *p = n;
13 }
14
15 int main(int argc, char* argv[]){
16     int a;
17     int* p=&a;
18
19     print_var(a);
20     a = 53;
21     print_var(a);
22     print_pointer(p);
23     set_pointer(p, 42);
24     print_pointer(p);
25     print_var(a);
26
27     return 0;
28 }
```

Exercice 5 J'affiche mon code source

Écrire un programme qui ouvre son fichier de code et lit caractère par caractère ce dernier pour tout afficher sur la sortie standard. Ceci devrait donc produire un programme dont l'exécution a pour effet d'afficher le code source de ce même programme.

voir la structure FILE, fopen, fgetc, putchar et le caractère de fin de fichier EOF.