Série 0x09

Pointeurs et tableaux

INFO1-TIN — 3 mars 2026, 7efbe5

Exercice 1

Que fait ce programme ?

```c
#include <stdio.h>
#include <string.h>

int main(int argc, char *argv[]) {
    for (int i = 0; i < argc; ++i) {
        char buffer[100];
        if (strlen(argv[i]) > 100) {
            printf("Argument trop long: ignore\n");
            continue;
        }
        strncpy(buffer, argv[i], 100);
        for (int j = 0; j < strlen(buffer); ++j)
            if (buffer[j] >= 'a' && buffer[j] <= 'z')
                buffer[j] -= 'a' - 'A';
        char *p = buffer;
        while (*p != '\0') {
            printf("%c", *p);
            p++;
        }
        puts("");
    }
}
```

Exercice 2 : Tableaux

Considérez les déclarations suivantes :

```c
int a[10];
int b[10][10];
int c[] = {1, 2, 3, 4, 5};
char d[] = "Hello";
int *e[10];
int (*f)[10];
```

(a) Quelle est la taille en mémoire de chaque déclaration ?

(b) Quelle est la différence entre `sizeof(a)`, `sizeof(*a)` et `sizeof(a[1])` ?

(c) Quelle est la différence entre `sizeof(d)`, `strlen(d)` ? Et comment est-ce que `strlen` fonctionne ?

(d) Pouvez-vous implémenter le comportement de `strlen` ?

(e) Comment afficher le contenu de `b` sous forme de matrice ligne colonne ?

Exercice 3

Pointeurs et fonctions

(a) Quelle est la différence entre `int *p` et `int* p` ?

(b) Quelle est la différence entre `int *p` et `int p[10]` ?

(c) Pour la déclaration `int a[10]`, si j'écrit `a[10] = 42`, est-ce que le compilateur va générer une erreur ? Et que se passe-t-il ?

Exercice 4

Passage par adresse et par valeur

(a) Que signifie ce prototype de fonction `void f(int *p)` ?

(b) Que signifie ce prototype de fonction `void f(int a[])`, comparez avec la question précédente.

(c) Considérant le code suivant, est-ce que le code est fonctionnel ? Que voyez-vous à l'écran si vous l'exécutez ?

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

size_t f(int a[]) {
    return sizeof(a);
}

int main() {
    int a[10];
    printf("%ld\n", sizeof(a));
    printf("%ld\n", f(a));
}
```

(d) Si le code est modifié avec ce prototype de fonction `void f(int a[10])`, comparez avec la questions précédente.

(e) Comment convenablement écrire la fonction `f` pour qu'elle s'adapte à n'importe quelle taille de tableau ? (indice, il faut deux paramètres)

(f) Comment modifier le prototype de `f` pour que le tableau puisse être parcouru mais qu'il ne puisse pas être modifié ?