

Pour vous préparer aux examens écrits, on vous propose de réaliser les exercices suivants sur papier avec un crayon et une gomme. Vous pouvez utiliser des feuilles de brouillon.

Exercice 1

Écrire une fonction qui reçoit en paramètre une chaîne de caractères et qui transforme les minuscules en majuscules dans cette chaîne. Ne pas utiliser la fonction standard `toupper`.

```
void to_upper(char *str) {
    int i = 0;
    while(str[i] != '\0') {
        if (str[i] >= 'a' && str[i] <= 'z')
            str[i] -= 'a' - 'A';
        i++;
    }
}
```

Exercice 2

Sans utiliser `strlen`, écrire une fonction qui calcule la longueur d'une chaîne de caractère.

```
size_t strlen(char *str) {
    size_t size = 0;
    while(str[size] != '\0') {
        size++;
    }
    return size;
}
```

Exercice 3

Écrire une fonction qui retourne une valeur aléatoire entière entre `a` et `b`, vous pouvez utiliser la fonction `rand()`.

```
int rand_range(int a, int b) {
    return rand() % (b - a + 1) + a;
}
```

Exercice 4

Écrire une fonction qui échange deux entiers passés par référence.

```
void swap(int *a, int *b)
{
    int tmp = *a;
    *a = *b;
    *b = tmp;
}
```

Exercice 5

Écrire une fonction qui reçoit trois valeurs réelles en paramètres et qui les trie dans l'ordre croissant. Par exemple si vous avez $a = 15, b = 23, c = 4$ après l'appel de fonction, ces valeurs vaudront : $a = 4, b = 15, c = 23$. Vous pouvez utiliser la fonction `swap` écrite plus haut pour échanger les valeurs.

```
void sort_3(double *a, double *b, double *c)
{
    if (*a > *b) swap(a, b);
    if (*a > *c) swap(a, c);
    if (*b > *c) swap(b, c);
}
```

Exercice 6

Écrire une fonction qui retourne la moyenne de trois valeurs réelles reçues en paramètres.

```
double mean_3(double a, double b, double c)
{
    return (a + b + c) / 3;
}
```

Exercice 7

Écrire une fonction qui affiche `version` sur la sortie standard si un argument `--version` est présent dans les arguments du programme. Cette fonction aura le prototype suivant, et la fonction `main` est donnée.

```
void parse_arguments(int argc, char *argv[])
{
    for (int i = 0; i < argc; i++)
        if (strcmp(argv[i], "--version") == 0)
            printf("Version\n");
}

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

Exercice 8

Écrire une fonction qui retourne la valeur minimale d'un tableau de réels. Le prototype est le suivant :

```
double min(double array[], size_t size);
```

```
double min(double array[], size_t size) {  
    double min = array[0];  
    for (int i = 1; i < size; i++)  
        if (array[i] < min)  
            min = array[i];  
    return min;  
}
```