



Systèmes Embarqués 1 & 2

Classes T-2/I-2 // 2017-2018

a.09 – C - Basics *Exercices*

Exercice 1

On va coder un algorithme de cryptage très simple: on choisit un décalage (par exemple 5). Un a sera remplacé par un f, un b par un g et un c par un h, etc. On ne cryptera que les lettres majuscules et minuscules sans toucher ni à la ponctuation, ni à la mise en page.

On demande de

- (a) Déclarer un tableau de caractères mess initialisé avec un message en clair
- (b) Écrire une procédure crypt de cryptage d'un caractère qui sera passé par valeur et retournera le caractère chiffré
- (c) Écrire le main qui activera crypt sur l'ensemble du message et imprimera le résultat

Exercice 2

Développer un petit programme permettant d'extraire le nom et les arguments des lignes de commande ci-dessous

- (a) "open arg1 arg2 arg3"
- (b) "close param1 param2 "
- (c) "read p1"
- (d) "write a1 a2 a3 a4 "

Exercice 3

Le code suivant implémente une fonction de chiffrement pour des chaînes de caractères. A partir de ce code, écrire en C la fonction de déchiffrement.

```
#define DECAL_MIN    5
#define DECAL_MAJ    9

// fonction de cryptage
char crypt (char p)
{
    if ((p >= 'a') && (p <= 'z')) {
        p += DECAL_MIN;
        if (p > 'z') p -= 26;
    } else if ((p >= 'A') && (p <= 'Z')) {
        p += DECAL_MAJ;
        if (p > 'Z') p -= 26;
    }
    return p;
}
```

**Exercice 4**

Traduire en langage ANSI-C le code assembleur ci-dessous en tenant compte des 3 variables globales « nbr », « i » et « res », « nbr » étant déjà initialisée. Indiquer le type de ces 3 variables.

```
ldr    r1, =i
mov     r0, #0
strb    r0, [r1]
ldr     r2, =res
str     r0, [r2]
b       test
trait:  ldrb   r0, [r1]
        ldr   r3, [r2]
        add   r0, r0, r3
        str   r0, [r2]
        ldrb  r0, [r1]
        add   r0, #1
        strb  r0, [r1]
test:   ldr    r3, =nbr
        ldrb   r3, [r3]
        cmp    r0, r3
        blo    trait
```