

Université Paris-Est Créteil
Faculté de Droit - IUT de Sénart-Fontainebleau
2020/2021
Premier semestre – Première année
Master Droit du Numérique – Informatique et Droit
Programmation structurée

PREMIÈRE SESSION – Mardi 12 janvier 2021

3 heures

Seules les notes manuscrites et les impressions de programmes portant le nom de l'étudiant (dans le programme et sur chaque page) sont permis à titre de documentation.

Aucun matériel électronique n'est autorisé.

Les exercices sont indépendants mais devront être présentés dans l'ordre.

Exercice 1.- Écrire un programme C qui permet d'entrer un certain nombre de réels positifs, en terminant par la valeur -1. Le programme doit alors afficher le nombre de réels entrés, la moyenne de ceux-ci (avec deux décimales), la somme de leurs cubes et la somme des inverses des réels non nuls.

Exercice 2.- (**Disparitions**)

- 1^o) Écrire une fonction C à un argument entier naturel et à valeur entier naturel, qui renvoie l'entier avec un chiffre sur deux de son développement décimal, le chiffre des unités étant présent.

[On a : $f(12345) = 135$ et $f(123456) = 246$.

On pourra utiliser la division euclidienne.]

- 2^o) Écrire un programme C qui permet de saisir des entiers naturels et d'afficher à l'écran un chiffre sur deux, le dernier chiffre étant celui des unités. On termine lorsque l'entier saisi est zéro.

[Un exemple de session est :

```
Entrer un entier : 12345
L'entier modifié est : 135
Entrer un entier : 123456
L'entier modifié est : 246
Entrer un entier : 0
Au revoir ! ]
```

Exercice 3.- (Base trois)

- 1^o) Écrire, en langage C, quatre fonctions `printZero()`, `printOne()` et `printTwo()` permettant d'écrire les chiffres 0, 1 et 2 respectivement sous la forme :

```
00000  11    22222
0  0    1      2
0  0    1      2
0  0    1    22222
0  0    1      2
0  0    1      2
00000  111   22222
```

- 2^o) Écrire un programme C qui demande un entier naturel et qui affiche celui-ci en base trois sur une enseigne verticale, c'est-à-dire en colonne en utilisant les fonctions ci-dessus, à l'envers, c'est-à-dire en commençant par le chiffre des unités.

[*Un exemple de session est :*

```
$ ./a.out
n = 7
```

```
11
 1
 1
 1
 1
 1
111
```

```
22222
  2
  2
22222
 2
 2
22222
$
```

```
]
```

DOCUMENTATION

Le programme C suivant demande un entier naturel et affiche son développement en base sept :

```
/* base7.c */

#include <stdio.h>

void main(void)
{
    int n, m, p;

    printf("n = ");
    scanf("%d", &n);
    m = 0; /* Initialisation du nombre saisi exprime en base sept */
    p = 1; /* Les puissances sept (en fait dix) successives */

    while (n != 0)
    {
        m = m + p*(n%7); /* ajout de l'unite d'ordre p */
        n = n/7; /* on deconstruit l'expression en base dix */
        p = 10*p; /* passage a la puissance superieure */
    }

    printf("En base sept n = %d\n", m);
}
```