

# M1102 : Algorithmique et Programmation

## Journée de Programmation Intensive

### Liste 1

**Exercice 1** Ecrire un algorithme qui permet de trouver le rayon, la hauteur ou le volume d'un cylindre connaissant les deux autres paramètres. Rappel :  $V = \pi r^2 h$ .

**Exercice 2** Ecrivez un programme qui prend en entrée un nombre entier  $n$  et qui donne en résultat la somme de toutes les puissances de 2 plus petites que  $n$ .

**Exercice 3** Ecrivez un programme qui produit l'affichage ci-dessous en fonction de  $n$  donné par l'utilisateur. Par exemple, pour  $n = 5$  (5 lignes et 5 colonnes) :

```
12345
23451
34512
45123
51234
```

**Exercice 4** Ecrivez un programme qui produit l'affichage ci-dessous en fonction de  $n$  donné par l'utilisateur. Par exemple, pour  $n = 5$  (5 lignes et 5 colonnes) :

```
11111
12222
12333
12344
12345
```

**Exercice 5** Ecrivez un programme qui produit l'affichage ci-dessous en fonction de  $n$  donné par l'utilisateur. Par exemple, pour  $n = 4$  (4 lignes) :

```
*
* *
* * *
* * * *
```

**Exercice 6** Ecrivez un programme qui produit l'affichage ci-dessous en fonction de  $n$  donné par l'utilisateur. Par exemple, pour  $n = 4$  (4 lignes) :

```

      *
    ***
  *****
*****

```

**Exercice 7** La suite de Fibonacci est une suite d'entiers dans laquelle chaque terme est la somme des deux termes qui le précèdent. Elle commence par les termes 0 et 1 et ses premiers termes sont : 0, 1, 1, 2, 3, 5, 8, 13, 21, etc. Ecrire un programme qui permet de calculer la moyenne des  $n$  premiers termes de la suite de Fibonacci, où  $n$  est fourni par l'utilisateur.

**Exercice 8** Écrire un programme qui prend en entrée un entier naturel  $n$  en base 10 et qui donne en résultat la représentation de  $n$  en base 11 (onze). Par exemple, la représentation en base 11 de l'entier 21 est 1A, car  $21 = 1 \times 11^1 + 10 \times 11^0$ .

**Exercice 9** Construire un programme pour jouer au jeu du plus ou moins. Cela s'agit du jeu où l'utilisateur doit deviner un nombre choisi par son opposant. Ici, on suppose que l'utilisateur jouera contre l'ordinateur. Les instructions sont comme suit :

Pour commencer l'ordinateur choisi au hasard un nombre compris entre 1 et 100. L'utilisateur doit alors deviner ce nombre comme ceci : L'utilisateur propose un nombre. L'ordinateur lui dit s'il est trop petit ou trop grand, et ainsi de suite jusqu'à ce que l'utilisateur aie trouvé le bon nombre.

Exemple d'exécution :

```

Quel est le nombre ?
50
C'est trop petit !

```

```

Quel est le nombre ?
75
C'est trop grand !

```

```

Quel est le nombre ?
60
Félicitations, vous avez trouvé le nombre mystère !!!

```

**Exercice 10** Écrire un programme qui prend en entrée un entier naturel  $n$  et retourne cet entier écrit à l'envers. Par exemple, si  $n = 1234$ , le programme devra retourner  $n = 4321$ . **Attention** : l'entrée s'agit bien d'un entier, vous n'avez pas le droit de le convertir en chaîne de caractère pour en suite lire ses éléments à partir de la fin. Tuyau : penser à utiliser les notions de quotient et de reste de la division entière.

**Exercice 11** Écrire un programme qui affiche le triangle de Pascal jusqu'au rang  $N$  donné par l'utilisateur. Pour le calcul de la ligne courante ( $n$ ), on s'appuiera sur la ligne précédente :

- $t[0] = 1$
- $t[i] = t_{precedent}[i - 1] + t_{precedent}[i]$ , pour tout  $i \in [1; n - 2]$
- $t[n - 1] = 1$

Par exemple, pour  $N=5$  nous avons :

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1

```

**Exercice 12** Ecrivez un programme qui lit deux listes d’entiers fournies par l’utilisateur et qui en suite calcule le “bili-bili” des deux listes. Pour calculer le bili-bili de deux listes d’entiers, il faut multiplier chaque élément d’une liste par chaque élément de l’autre, puis additionner le tout et à la fin faire la somme des chiffres du résultat. Par exemple si l’on a :

```

liste1 = [4, 8, 7, 12]
liste2 = [3, 6]

```

le bili-bili sera 18, car  $3 * 4 + 3 * 8 + 3 * 7 + 3 * 12 + 6 * 4 + 6 * 8 + 6 * 7 + 6 * 12 = 279$  et  $2 + 7 + 9 = 18$ .

**Exercice 13** Le Code de César permet de décaler les lettres d’une phrase d’un rang dans l’alphabet (le A devient B, le B devient C, etc). Construire un algorithme “codeur” permettant de résoudre le problème suivant :

- Entrée : une chaîne de caractères
- Sortie : la chaîne cryptée selon le code de César

**Exercice 14** Construire un algorithme “décodeur” permettant de résoudre le problème suivant :

- Entrée : une chaîne de caractères cryptée selon le code de César
- Sortie : la chaîne originale décryptée

**Exercice 15** Ecrire un algorithme permettant de trouver un nombre entier à 4 chiffres  $abcd$  et tel que  $4 \times abcd = dcba$ .