



Module : Architecture des ordinateurs et Algorithmique

1^{ère} API / S1 / Année 2018/2019

Feuille de Travaux Dirigés N° 5

Exercice 1

Écrire un algorithme qui demande la saisie d'un nombre entier N, et qui indique ensuite si N est « pair » ou « impaire » :

1. En considérant le reste de la division de N par 2.
2. En considérant le chiffre des unités de N.

Exercice 2

Un triangle est « équilatéral », si ses trois côtés sont égaux. Un triangle est « isocèle », si deux de ses côtés sont égaux. Un triangle est « scalène » si ses trois côtés sont deux à deux distincts. Écrire un algorithme qui lit trois entiers représentant les trois côtés d'un triangle, et qui indique ensuite par un message sa nature (équilatéral, isocèle ou scalène).

Exercice 3

Écrire un algorithme qui permet de résoudre une équation de second degré dont les coefficients ont été saisis au clavier (le coefficient du plus haut degré peut être nul).

Exercice 4

Les personnes historiques figurant sur les billets de banque aux États-Unis sont répertoriées dans le tableau suivant :

Individual	Amount
George Washington	\$1
Thomas Jefferson	\$2
Abraham Lincoln	\$5
Alexander Hamilton	\$10
Andrew Jackson	\$20
Ulysses S. Grant	\$50
Benjamin Franklin	\$100

Écrire un algorithme qui saisit le montant d'un billet de banque. Ensuite, l'algorithme devrait afficher le nom de la personne qui apparaît sur le billet de banque du montant entré. Un message d'erreur approprié doit être affiché si le montant saisi ne figure pas dans le tableau.

Exercice 5

1. Écrire un algorithme qui saisit un nombre entier N , et qui affiche dans l'ordre croissant tous les nombres pairs $\leq N$.
2. Écrire un algorithme qui saisit un nombre entier N , et qui affiche dans l'ordre décroissant tous les nombres pairs $\leq N$.
3. Écrire un algorithme qui saisit un nombre entier N , et qui calcule et affiche la somme de tous les nombres impairs $\leq N$.

Exercice 6

Écrire un algorithme qui force l'utilisateur à saisir un multiple de 7.

Exercice 7

Écrire un algorithme qui affiche le $n^{\text{ème}}$ terme de la suite numérique définie par :

$$u_0 = 1, \text{ et } u_{n+1} = 2(2u_n + 1) / (u_n + 3), \text{ pour } n \geq 0$$

Exercice 8

1. Écrire un algorithme qui affiche dans l'ordre croissant, tous les nombres entiers de trois chiffres qui sont divisibles par la somme de leurs chiffres.
2. Écrire un algorithme qui permet de tester la propriété suivante : « Dans toute séquence de 18 nombres entiers consécutifs et formés de trois chiffres, il y a au moins un qui est divisible par la somme de ses chiffres ».

Exercice 9

1. Écrire un algorithme qui affiche le $n^{\text{ème}}$ terme de la suite de Fibonacci définie par $F_0 = F_1 = 1$, et $F_n = F_{n-1} + F_{n-2}$ pour $n \geq 2$.
2. Modifier l'algorithme précédent pour qu'il affiche le premier terme n tel que F_n soit plus grande que 1000.
3. Modifier l'algorithme précédent pour qu'il affiche le premier terme n pour lequel le chiffre des unités de F_n est égal à 6.

Exercice 10

Écrire un algorithme qui saisit un nombre $N < 10$ et qui affiche le motif suivant, par exemple si $N = 5$:

1

22

333

4444

55555