
Série №3 TD Algorithme 1

Objectifs : Les structures de contrôles Résolution de problèmes algorithme.

Rappel et Avertissement: La présence et la préparation des TD/TP est obligatoire. Celui ou Celle qui ne respecte pas les règles sur deux séances, ne fera pas d'évaluations ni intégrations d'algorithme.

Exercice 1 :

Écrire un algorithme qui permet de calculer les racines de l'équation du second degré suivante : $ax^2 + bx + c = 0$. Comme entrées, l'utilisateur doit introduire les facteurs a, b et c.

Exercice 2 :

Écrire un algorithme intitulé TOUCHE, qui affiche selon le cas, la nature du caractère (consonne, voyelle, chiffre ou symbole) correspondant à une touche saisie. On considère que le clavier est verrouillé en minuscule.

Exercice 3 :

Écrire un algorithme qui permet d'afficher le nombre de jour d'un mois donné.

NB : Une année bissextile est une année comptant 366 jours au lieu de 365, c'est-à-dire une année comprenant un 29 février (exemple 2012) sont bissextiles les années :

Divisibles par 4 mais non divisibles par 100 divisibles par 400.

Exercice 4 :

Écrire un algorithme qui demande d'entrer un nombre entre 1 et 7, et donne le nom ; du jour correspondant (samedi, dimanche, ...) en utilisant la conditionnelle multiple.

Exercice 5 :

Écrire un programme qui, à partir de la saisie de deux réels et un opérateur affiche le résultat après exécution de l'opération choisie.

Exemple :

Si les entiers sont 14 et 2 et l'opérateur est "+" alors le résultat affiché est $14.00 + 2.00 = 16.00$

Pour les mêmes entiers et si l'opérateur est "/" alors le résultat affiché est $14.00 / 2.00 = 7.00$

Exercice 6 :

Écrire un algorithme permettant à partir d'un menu affiché, d'effectuer la somme, le produit ou la moyenne de trois nombres. Nous appelons menu, l'association d'un numéro séquentiel aux différents choix proposés par un programme.

Exercice 7 :

Écrire un algorithme qui demande à l'utilisateur un nombre compris entre 1 et 3 jusqu'à ce que la réponse convienne.

Exercice 8 :

Écrire un algorithme qui demande un nombre compris entre 10 et 20, jusqu'à ce que la réponse convienne. En cas de réponse supérieure à 20, on fera apparaître un message : « Plus petit ! », et inversement, « Plus grand ! » si le nombre est inférieur à 10.

Exercice 9 :

Écrire un algorithme qui demande un nombre de départ, et qui ensuite affiche les dix nombres suivants. Par exemple, si l'utilisateur entre le nombre 17, l'algorithme affichera les nombres de 18 à 27.

Exercice 10 :

Écrire un algorithme qui demande un nombre de départ, et qui calcule la somme des entiers jusqu'à ce nombre. Par exemple, si l'on entre 5, l'algorithme doit calculer :

$$1 + 2 + 3 + 4 + 5 = 15$$

NB : on souhaite afficher uniquement le résultat, pas la décomposition du calcul.

Exercice 11 :

Écrire un algorithme qui demande un nombre de départ, et qui calcule sa factorielle.

NB : la factorielle de 8, notée 8!, vaut $1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8$

Exercice 12 :

Écrire un algorithme qui demande successivement 20 nombres à l'utilisateur, et qui lui dise ensuite quel était le plus grand parmi ces 20 nombres et affiche en quelle position a été saisie ce nombre :

Entrez le nombre numéro 1 : 12

Entrez le nombre numéro 2 : 14

etc.

Entrez le nombre numéro 20 : 6

Le plus grand de ces nombres

C'était le nombre numéro 2

Exercice 13 :

Réécrire l'algorithme précédent, mais cette fois-ci on ne connaît pas d'avance combien l'utilisateur souhaite saisir de nombres. La saisie des nombres s'arrête lorsque l'utilisateur entre un zéro.

Exercice 13 :

Écrire un algorithme qui permet de convertir un nombre entier en base décimale (base 10) en un nombre binaire (base 2). Nous utiliserons la méthode de division consécutive par 2.

NB : Contrôle de saisie : l'entier (en base 10) est strictement supérieur à 0.

Exercice 14 :

Écrire un algorithme en pseudo-code qui inverse le nombre entré par l'utilisateur, puis affiche le nombre inversé à l'écran.

Par exemple, si l'utilisateur entre

1. 2022 en entrée, 2202 est affiché en sortie
2. 678 en entrée, 876 est affiché en sortie.

Nous utilisons les opérateurs modulo (mod), division entière (div), la multiplication et l'addition dans le programme pour obtenir les chiffres d'un nombre.