

Introduction à l'algorithmique et à la programmation

Fiche de TD N°1/Semestre 1

Questions de cours

1. REPONDRE PAR VRAI OU FAUX

1. L'algorithmique définit la façon d'élaborer des algorithmes.
2. Un algorithme est une branche de l'informatique.
3. La programmation consiste à traduire un algorithme dans un langage parlé couramment par les hommes.
4. Une donnée est soit en entrée, soit en sortie et rien d'autre.
5. Le fonctionnement d'un ordinateur fait intervenir la mémoire et un exécutant.
6. Un programme est moins détaillé qu'un algorithme.
7. Une instruction est une donnée d'un algorithme.
8. Dans les expressions arithmétiques, les fonctions sont prioritaires sur les parenthèses.
9. Les constantes sont superflues (inutiles) dans un algorithme.
10. Les structures de contrôle définissent l'ordre de saisie des données d'un algorithme.

2. ENCERCLER LA REPONSE LA PLUS COMPLETE ET CORRECTE

1. Un algorithme utilisant des constantes:
 1. est plus lisible pour l'être humain
 2. est plus lisible et plus facile à modifier si la valeur de la constante doit être mise à jour.
 3. est non seulement moins lisible, mais plus difficile à modifier.
2. Les entrées et sorties d'un algorithme :
 1. sont nécessairement du même nombre.
 2. sont nécessairement d'un nombre différent.
 3. doivent être clairement identifiées dans l'écriture complète d'un algorithme
3. L'algorithme est destiné à :
 1. être exécuté par une machine bien précise
 2. être lu par un humain
 3. être à la fois exécuté par une machine bien précise et lu par l'homme

3. REPONDRE AUX QUESTIONS SUIVANTES

1. Enumérer les étapes de résolution d'un problème à l'aide de l'ordinateur.
2. A qui servent les boucles ?
3. Quel type de structure de contrôle faut-il utiliser pour répéter un nombre X donné de fois, une série d'instructions.
4. Citez trois fonctions et trois opérateurs arithmétiques.
5. Dans quel ordre exécutons des opérateurs de même ordre de priorité ?
6. A quoi servent les parenthèses dans une expression arithmétique ?
7. Enumérer et donnez les tables de vérité des opérateurs logiques utilisés en algorithmique.
8. Donnez la syntaxe et la sémantique de la structure **TANT QUE**.
9. Qu'est-ce qu'un compilateur ?
10. Quel est l'ordre de priorité des opérateurs logiques ?
11. Quand faut-il utiliser la structure SELON ?
12. Un ami vous dit qu'au lieu de perdre du temps à écrire un algorithme, lui il préfère écrire directement un programme. Qu'allez-vous lui dire ?

4. ASSOCIEZ CES DEUX COLONNES

Colonne 1	Colonne 2
1. algorithme	a. la valeur ne change pendant l'exécution
2. compilateur	b. un emplacement de la mémoire
3. constante	c. langage de programmation
4. algorithme d'Euclide	d. algorithmique
5. Structure de contrôle	e. propriété d'un algorithme
6. expression logique	f. formulation de la solution d'un problème
7. discipline de l'informatique	g. POUR ... FAIRE
8. terminaison	h. s'évalue à Vrai ou Faux
9. variable	i. procédé de calcul
10. Pascal	j. est un programme informatique

NB : Vous pouvez simplement donner la correspondance parfaite un à un entre les nombres de la colonne 1 et les lettres de la colonne 2.

5. QUE FONT LES ALGORITHMES SUIVANTS

1.

```
ALGORITHME afficher
DEBUT
    Ecrire("La valeur de 3*5 est ", 3*5)
FIN
```

2.

```
ALGORITHME MOYENNE
CONST C1=2
      C2=3
      C3=1
VAR
    N1, N2, N3 : REEL
    MOY : REEL
DEBUT
    {Affichage message : Invitation de l'utilisateur à introduire des données}
    ECRIRE(" Donner trois valeurs réelles ")
    {Saisie des notes}
    LIRE(N1, N2, N3)
    {Calcul de la moyenne}
    MOY ← (N1*C1+N2*C2+N3*C3) / (C1+C2+C3)
    {Affichage du résultat sur l'écran}
    ECRIRE(" La moyenne est = ", MOY)
FIN
```

3. Les noms de ces algorithmes sont-ils évocateurs ?
4. A qui servent les textes placés entre les accolades { ...} ?

Exercices

Exercice 1

Soit l'algorithme suivant :

```
Algorithme Equation2D
VAR
  a, b, c : REEL
  delta : REEL
DEBUT
  a <- 2
  b <- -3
  c <- 1
  delta <- b*b-4*a*c
  Ecrire("Le discriminant est = ", delta)
Fin
```

1. Combien de variables a-t-on dans cet algorithme ?
2. Le nom de cet algorithme est-il évocateur ?
3. Quelle est la valeur de la variable delta de cet algorithme à l'issue de l'exécution ?
4. Qu'affiche à l'écran l'exécution de cet algorithme ?

Exercice 2

On veut écrire un algorithme qui lit une valeur numérique x et affiche : « POSITIF » ou « NEGATIF » ou « NUL » selon que la valeur de x est positive ou négative ou nulle.

1. Quelles sont les entrées et les sorties de l'algorithme à écrire ?
2. Ecrivez l'algorithme complet.

Exercice 3

L'amortissement linéaire simple est une méthode d'amortissement où le montant de l'amortissement est constant et égal à la valeur initiale du bien divisée par le nombre d'années (durée de vie du bien). On veut écrire un algorithme permettant de calculer le montant de l'amortissement annuel d'un bien. Par exemple pour un bien de 4 500 000 FCFA, amorti en 5 ans, le montant annuel de l'amortissement sera de 4 500 000 / 5 soit 900 000 FCFA.

- Identifiez les entrées et sorties de votre algorithme ?
- Ecrivez l'algorithme.
- Votre algorithme utilise-t-elle de variables intermédiaires (appelées aussi variables de calcul) ?

Exercice 4

On veut écrire un algorithme permettant de calculer et d'afficher le salaire net d'un employé. Sachant que :

- Le salaire net = Salaire brut – valeur de l'impôt – valeur de CNPS
- Salaire brut = (Salaire de base + Prime de technicité + Prime de transport + Prime des enfants)*Taux de travail
- Taux de travail = Nombre de jours travaillés / 26
- Prime des enfants = Prime d'un enfant * Nombre d'enfants
- Valeur de l'impôt = Taux de l'impôt * Salaire brut
- Valeur CNPS = Taux de CNPS * Salaire brut

- Taux CNPS = 26,5%
- Taux de l'impôt = 2%

On vous demande de :

1. Identifier les entrées et les sorties de votre algorithme
2. Votre algorithme utilise-t-il des constantes ? si oui dites lesquelles ?
3. Votre algorithme utilise-t-il des variables de calcul ? si oui, dites lesquelles ?
4. Ecrivez l'algorithme.

Exercice 5

Evaluer les expressions logiques suivantes, avec $(a, b, c, d) = (2, 3, 5, 10)$ et $(X, Y) = (V, F)$.

1) $(a < b) \wedge (a < c)$	2) $\neg ((a < b) \wedge (a < c))$	3) $\neg (a < b) \wedge (a < c)$
4) $(a < c) \wedge (c = d/2)$	5) $(d / a = c) = Y$	6) $(d / c = b) = Y$
7) $(d / c = b) = X$	8) $(a < b) \wedge (d < c)$	9) $(a < b) \wedge (d < c) = X$

Exercice 6

On veut écrire un algorithme qui sera utilisé pour calculer le montant des heures supplémentaires d'un employé, sachant que le prix unitaire d'une heure est donné selon le barème suivant :

- Les 39 premières heures sont sans supplément,
- De la 40^{ème} à la 44^{ème} heure sont majorées de 50%,
- De la 45^{ème} à la 49^{ème} heure sont majorées de 75%,
- De la 50^{ème} heure et au-delà, sont majorées de 100%.

On vous demande :

1. Identifier les entrées et les sorties de votre algorithme
2. Votre algorithme utilise-t-il des constantes ? si oui dites lesquelles ?
3. Votre algorithme utilise-t-il des variables de calcul ? si oui, dites lesquelles ?
4. Ecrivez l'algorithme.

Exercice 7

Ecrire l'algorithme CONTRAT qui permet d'aider une compagnie d'assurance à prendre une décision concernant les demandes d'affiliation en se basant sur les critères suivants :

DECISION \ CRITERE	AGE	Bonne santé	Accident
Contrat A	≤ 30	OUI	NON
Contrat B	> 30	OUI	OUI
Contrat refusé	-	NON	OUI
Expertise demandée	-	OUI	OUI

Exercice 8

- 1) Ecrire un algorithme qui lit la vitesse d'un véhicule et qui affiche :
 - a. « OK » si la vitesse est inférieure ou égale à 80Km/h
 - b. « EXCES » si la vitesse est supérieure à 80Km/h.
- 2) Modifiez l'algorithme pour qu'il lise un certain nombre de vitesse d'automobile et produire (c'est-à-dire afficher à l'écran) :
 - a. Le nombre de vitesse en dessous ou égales à 80Km/h

- b. Le nombre de vitesse au-dessus de 80Km/h
 - c. Le taux des infractions (% de vitesses supérieures à 80Km/h).
- La fin de la saisie des vitesses sera spécifiée par la saisie d'une vitesse négative.

Exercice 9

Ecrire un algorithme qui permet d'afficher les diviseurs d'un nombre.

Exercice 10

Ecrire un algorithme permettant de déterminer si un nombre entier est parfait ou non. Un entier est dit parfait s'il est égal à la somme de ses diviseurs. Exemple : $6 = 3+2+1$, donc 6 est un nombre entier parfait.

Exercice 11

On veut écrire un algorithme permettant de tirer un nombre entier positif ou nul au hasard, ensuite, l'utilisateur aura 10 chances pour deviner la valeur tirée. S'il réussit à deviner dans les 10 essais, il (l'utilisateur gagne le jeu).

Pour guider l'utilisateur, votre algorithme devra afficher un des trois messages suivants :

- « Vous êtes en dessous » lorsqu'une valeur inférieure est proposée,
- « Vous êtes au-dessus » lorsqu'une valeur supérieure est proposée.

Après 10 tentatives infructueuses, l'utilisateur perd le jeu.

- a) Ecrivez cet algorithme.
- b) Modifiez l'algorithme afin que le nombre d'essais (10) soit spécifié interactivement.

Exercice 12

On appelle persistance d'un nombre, le nombre d'itérations (répétitions) nécessaires pour le réduire à un seul chiffre, en multipliant tous les chiffres qui le composent et en recommençant avec le résultat obtenu. Par exemple la persistance de 6788 est 6 ; car :

$$6 \times 7 \times 8 \times 8 = 2688 \leftarrow \text{1}^{\text{ère}} \text{ itération}$$

$$2 \times 6 \times 8 \times 8 = 768 \leftarrow \text{2}^{\text{ème}} \text{ itération}$$

$$7 \times 6 \times 8 = 336 \leftarrow \text{3}^{\text{ème}} \text{ itération}$$

$$3 \times 3 \times 6 = 54 \leftarrow \text{4}^{\text{ème}} \text{ itération}$$

$$5 \times 4 = 20 \leftarrow \text{5}^{\text{ème}} \text{ itération}$$

$$2 \times 0 = 0 \leftarrow \text{6}^{\text{ème}} \text{ itération}$$

- a) Ecrire un algorithme qui lit un nombre et qui affiche la liste de droite à gauche des chiffres qui le composent. Exemple : $539 \leftarrow 9, 3, 5$.
- b) Ecrire un algorithme qui lit un nombre et qui affiche le produit des chiffres qui le composent. Exemple : $247 \leftarrow 56$.
- c) Ecrire un algorithme qui entre un nombre et qui affiche la persistance de ce nombre.