



– Algorithmique I – Travaux Dirigés – Série N°1

Exercice 1 : Algorithmes en langage naturel

On peut considérer un algorithme comme une machine fonctionnant en trois étapes :

1. Introduire les données nécessaires : Les **entrées**.
2. Exécuter séquentiellement des instructions sur ces données : Les **traitements**.
3. Afficher les résultats obtenus : Les **sorties**.

Les entrées (données) et les sorties (résultats) forment la partie **déclarations** d'un algorithme ;

La partie **traitements** (actions) contient la liste des instructions (opérations).

Décrire la partie de déclarations, puis la partie d'actions pour les problèmes suivants :

1. Déterminer si un nombre entier **N** est pair ou impair.
2. Résolution de l'équation du premier degré suivante dans **R** : $aX + b = 0$.
3. Résolution de l'équation du second degré suivante dans **R** : $aX^2 + bX + c = 0$.
4. Détermination si un nombre entier **N** est premier ou pas. Un nombre est dit premier s'il n'admet que deux diviseurs : 1 et lui-même.

Exercice 2 : Déroulement d’algorithmes

Q1) Que fait cet algorithme ?

Algorithme Solution1

Var A, B : Entier // ≡ Soient A et B deux nombres entiers.

Début

Lire(A)	// ≡ Donner une valeur initiale à A.
Lire(B)	// ≡ Donner une valeur initiale à B.
A ← A + B	// ≡ Calculer A + B et remplacer A par la valeur trouvée.
B ← A – B	// ≡ Calculer A – B et remplacer B par la valeur trouvée.
A ← A – B	// ≡ Calculer A – B et remplacer A par la valeur trouvée.
Ecrire(A, B)	// ≡ Afficher les nombres A et B.

Fin

Q2) En souhaitant arriver au même résultat, un(e) étudiant(e) en 1^{ière} année de licence (MI) a proposé la solution suivante :

Algorithme Solution2

Var A, B : Entier // ≡ A et B deux nombres entiers.

Début

Lire(A)	// ≡ Donner une valeur initiale à la variable entière A.
Lire(B)	// ≡ Donner une valeur initiale à la variable entière B.

```

A ← B      // ≡ Donner la valeur de B à A.
B ← A      // ≡ Donner la valeur de A à B.
Ecrire(A, B) // ≡ Afficher les valeurs des deux variables A et B.
Fin

```

Dérouler cet algorithme et vérifier s'il permet, réellement, de faire la même chose que l'algorithme **Solution1** ? Si la réponse est non, corriger le.

Q3) Les deux algorithmes marchent-ils aussi pour les autres types de données standards : Réel, Caractère, Chaîne et Booléen ?Quelle conclusion proposez-vous ?

Exercice 3 : Correction d'algorithmes (Vérification de la syntaxe et contrôle de types)

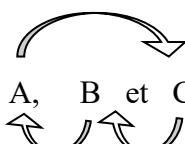
Q1) Trouvez les erreurs dans les deux algorithmes ci-dessous, en précisant pour chacune son type : Erreur de syntaxe, types incompatibles, opération indéfinie, variable non-déclarée, identifiant mal-orthographié, ...

Q2) Proposer des corrections possibles en justifiant votre réponse.

Algorithme test1 Const C= 10 Var A : Entier Début Lire(A) Lire(B) A ← A ² B← A mod C C ← A – B Ecrire(A, B, C) Fin	Algorithme test2 Var A : Entier B : Réel 2C : Caractère St : Chaine Début Lire(A) Lire(B) A ← B / 2 A ← A + B C ← ‘informatique’ St ← A+C Ecrire(A, B) Fin
--	---

Exercice 4 : Écriture d'algorithmes simples

1. Écrire un algorithme qui permet de calculer la somme (addition) de deux entiers A et B.
2. Modifier l'algorithme précédent pour qu'il calcule aussi le produit (la multiplication), la soustraction et la division (on suppose pour l'instant que $B \neq 0$) des deux entiers A et B.
3. Écrire un algorithme qui demande un nombre entier à l'utilisateur, puis calcule et affiche son carré.
4. Étant données 3 variables de type Réel : A, B et C, proposer un algorithme pour les permutez circulairement, en transférant les valeurs initiales de C à B, de B à A, et de A à C.



permuter circulairement, en transférant les valeurs initiales de C à B, de B à A, et de A à C.