

Algorithmes - Exercices de base

Voici une série de **10 exercices d'algorithmique sans code**, spécialement conçus pour t'entraîner à **réfléchir avant de coder**. Ces activités te permettront de t'exercer à **décomposer un problème**, identifier les variables, choisir les bonnes structures de contrôle, et formaliser des solutions pas à pas.



Exercices d'algorithmique (sans code)

Objectifs pédagogiques

- Développer la capacité à concevoir un algorithme avant de programmer.
- Savoir identifier les **variables utiles**.
- Choisir entre une **condition**, une **boucle** `for` ou une **boucle** `while`.
- Décrire les **étapes d'un raisonnement logique** à l'aide de phrases claires ou pseudo-code.
- Favoriser une démarche méthodique et structurée.

12
34

Exercice 1 : Nombre pair ou impair

Contexte : Tu reçois un nombre entier.

Consigne : Écris un algorithme qui permet de déterminer si ce nombre est pair ou impair.



Indice : Quelle opération mathématique permet de vérifier la parité ?



Exercice 2 : Accès à une fête

Contexte : Un videur vérifie si une personne peut entrer à une fête. Les règles :

- Il faut avoir **au moins 18 ans**

- Ou être **accompagné par un adulte**

Consigne : Écris un algorithme qui prend en compte ces règles et dit si la personne peut entrer.



Exercice 3 : Compter jusqu'à N

Contexte : On te donne un nombre entier positif **N**.

Consigne : Écris un algorithme qui affiche les nombres de 1 à N.

À quel moment sais-tu que tu dois t'arrêter ?



Exercice 4 : Compter les éléments d'une liste de prix

Contexte : Une liste de prix d'articles achetés est donnée. **Consigne** : Écris un algorithme qui compte le nombre d'éléments de cette liste.

Comment répètes-tu une action sur tous les éléments d'une liste ?



Exercice 4bis : Somme d'une liste de prix

Contexte : Une liste de prix d'articles achetés est donnée.

Consigne : Écris un algorithme qui calcule le **total à payer**.

Comment répètes-tu une action sur tous les éléments d'une liste ?



Exercice 5 : Compter à rebours

Contexte : Un feu d'artifice démarre après un compte à rebours.

Consigne : Écris un algorithme qui compte de 10 à 0 puis affiche "BOOM !".



Exercice 6 : Deviner un nombre

Contexte : Tu penses à un nombre entre 1 et 100.

Consigne : Écris un algorithme qui fait deviner ce nombre à un joueur, qui propose des réponses jusqu'à ce qu'il trouve.



Quand faut-il s'arrêter ? Que faire si le joueur se trompe ?



Exercice 7 : Élire un gagnant

Contexte : Trois candidats participent à une élection. On connaît leur nombre de votes respectifs.

Consigne : Écris un algorithme qui affiche le nom du gagnant.



Comment comparer plusieurs valeurs entre elles ?



Exercice 8 : Moyenne d'une série

Contexte : On a une liste de notes.

Consigne : Écris un algorithme qui calcule la moyenne et affiche si l'élève est "réussi" (moyenne $\geq 50\%$).



Comment additionner toutes les notes et combien y en a-t-il ?



Exercice 9 : Table de multiplication

Contexte : Tu veux afficher la table de multiplication d'un nombre choisi.

Consigne : Écris un algorithme qui affiche les produits de ce nombre multiplié par 1 jusqu'à 10.



Exercice 10 : Tant que ce n'est pas fini

Contexte : Un joueur peut entrer un mot de passe. Il peut recommencer **tant qu'il ne l'a pas trouvé**.

Consigne : Écris un algorithme qui demande un mot de passe jusqu'à ce qu'il soit correct.



Quelle est la condition d'arrêt ?



À retenir

Pour chaque problème algorithmique :

1. Lis attentivement l'objectif.
2. Identifie les données d'entrée et les résultats attendus.
3. Repère les actions à répéter, à tester ou à calculer.
4. Utilise des structures comme :
 - Variables (pour stocker une valeur)
 - Conditions (`si... sinon...`)
 - Boucles `for` (quand tu sais combien de fois répéter)
 - Boucles `while` (quand tu ne sais pas combien de fois, mais tu sais quand arrêter)