

Algorithmique et programmation Python

Mathématiques · Seconde
Contrôle n°1 · 55 min · sur 20

Nom et prénom

Classe

Date

NOTE

... / 20

CONSIGNES

- Durée : **55 minutes**. Le temps conseillé par exercice est indiqué à côté du barème.
- Le barème est donné : il dit où passer du temps.
- Écris tes réponses sur les lignes prévues. Une réponse hors des lignes reste comptée.

AVANT DE COMMENCER — CE QU'IL FAUT SAVOIR

- Un algorithme qui ne marche pas est presque toujours un raisonnement incomplet.
- L'affectation range une valeur ; elle n'affirme pas une égalité.
- En Python, l'indentation n'est pas une commodité de lecture : c'est la syntaxe.
- Dans `range(1, 11)`, la borne de droite est exclue.
- Avant d'écrire une boucle « tant que », il faut savoir quelle ligne rapproche de la sortie.
- Renvoyer et afficher ne sont pas la même chose.

Exercice 1 — Variables et indentation

4 pts 10 min

Comprendre affectation et indentation

Réponds à chaque question.

1. Que signifie exactement l'affectation `n = 5` ?

1 pt

2. Pourquoi `n = n + 1` a-t-il un sens en programmation ?

1 pt

3. À quoi sert l'indentation en Python ?

1 pt

4. Quelle différence y a-t-il entre `=` et `==` ?

1 pt

Exercice 2 — Les deux boucles

4 pts 11 min

Choisir et écrire une boucle

Réponds à chaque question.

1. Quand emploie-t-on une boucle ``for`` ?

1 pt

2. Quelles valeurs prend ``i`` dans ``for i in range(1, 11)`` ?

1 pt

3. Quand emploie-t-on une boucle ``while`` ?

1 pt

4. Que se passe-t-il si cette condition ne peut jamais devenir fausse ?

1 pt

Exercice 3 — Fonctions

4 pts 10 min

Écrire et comprendre une fonction

Réponds à chaque question.

1. Qu'est-ce qu'une fonction, en programmation ?

1 pt

2. Que fait l'instruction ``return`` ?

1 pt

3. Que fait ``print``, et en quoi diffère-t-elle ?

1 pt

4. Quelle question se poser avant de rendre une fonction ?

1 pt

Exercice 4 — Les algorithmes

5 pts 14 min

Connaître les algorithmes du programme

Réponds à chaque question.

1. Avec quoi initialise-t-on la recherche d'un maximum, et pourquoi pas avec 0 ? 1 pt

2. Quel est le principe de l'algorithme d'Euclide ? 1 pt

3. Déroule Euclide sur 48 et 18 : donne les restes successifs. 1 pt

4. Quel est le PGCD de 48 et 18 ? 1 pt

5. Jusqu'où suffit-il de tester les diviseurs pour savoir si n est premier ? 1 pt

Exercice 5 — Trouve l'erreur

3 pts 10 min

Repérer une erreur et la corriger

Chaque ligne est fautive. Écris ce qu'il fallait écrire.

1. « ``if n = 3:`` » 1 pt

2. « Dans le test de primalité, on écrit ``return True`` à l'intérieur de la boucle. » 1 pt

3. « Mon programme s'exécute sans message d'erreur : il est donc juste. » 1 pt