Corrigé et barème – Programmation en Python

Variables et affectations, types, conditions, boucles bornées et non bornées, fonctions (SNT de 2nde)

Durée : 60 min – Sans document – Sur 20 points · Sciences numériques et technologie (SNT) 2nde

Exercice 1 – Affectations et types *(4 pts)*

a) y = 18 / 2 = 9.0 : type float (/ donne toujours un flottant). (1)

b) x = 18 + 9.0 = 27.0 : type float. (1)

c) str(27.0) = '27.0', donc z = '27.01' : type str. (1)

d) $27 = 4 \times 6 + 3$: 27 // 4 = 6 et 27 % 4 = 3. (1)

Erreur fréquente — La division / produit un flottant même quand elle « tombe juste », et ce flottant se propage aux calculs suivants.

Exercice 2 – Instructions conditionnelles *(4 pts)*

a) $9 < 14$ est vrai : la fonction renvoie 5.5. (1)

b) $14 < 14$ est faux, $14 < 26$ est vrai : 7.9. (1)

c) $26 < 14$ et $26 < 26$ sont faux : on exécute le else, 11.2. (1)

d) On insère avant le else : `elif age >= 65: return 8.4` ; les cas 26 à 64 ans restent à 11.2 dans le else. (1)

À retenir — Avec des inégalités strictes, la valeur frontière bascule dans le cas suivant : il faut tester les bornes une par une.

Exercice 3 – Boucle bornée *(4 pts)*

a) range(1, 6) donne 1, 2, 3, 4, 5 : 5 tours. (1)

b) p vaut 1, 2, 6, 24, 120. (1)

c) Le produit $1 \times 2 \times 3 \times 4 \times 5 = 120$. (1)

d) Remplacer par `range(1, 10)` (10 exclu) : p = 362 880. (1)

Repère — Pour un produit, on initialise à 1 et non à 0, sinon le résultat reste nul.

Exercice 4 – Boucle non bornée *(4 pts)*

a) On cherche le nombre de mois pour atteindre 1 000 € : il n'est pas connu à l'avance. (1)

b) Il faut $250 + 45m \geq 1000$, soit $m \geq 750 / 45 \approx 16{,}7$: m = 17. (1)

c) $s = 250 + 45 \times 17 = 1\,015$. (1)

d) 750 / 45 n'est pas entier : s passe de 970 à 1 015 sans valoir 1 000, la boucle est infinie. (1)

Attention — Une condition d'arrêt fondée sur une égalité exacte risque de n'être jamais atteinte : on préfère une inégalité.

Exercice 5 – Fonctions *(4 pts)*

a) `def vitesse(d, t): return d / t` (1)

b) 126 / 1,5 = 84 : la fonction renvoie 84.0. (1)

c) `def maximum(a, b):` puis `if a >= b: return a`, sinon `return b`. (1)

d) La fonction affiche 6 mais n'a pas de return : elle renvoie None, qui est affecté à r. (1)

Le point clé — print montre un résultat à l'écran, seul return le transmet au programme.

Ce document est la propriété intellectuelle de Ben, professeur particulier (benprofparticulier.fr), protégée par le Code de la propriété intellectuelle (art. L. 111-1 et L. 122-4). Il est mis à disposition pour un usage personnel de révision. Toute reproduction, diffusion, mise en ligne ou revente, totale ou partielle, sans autorisation écrite est interdite et constitue une contrefaçon (art. L. 335-2).