

Corrigé et barème – Arithmétique

Divisibilité, division euclidienne, PGCD, nombres premiers et congruences — chapitre d'approfondissement, hors programme de l'option en Terminale

Durée : 120 min – Calculatrice interdite – Sur 20 points · Maths complémentaires (option Tle) Terminale

Exercice 1 – Divisibilité et division euclidienne

(4 pts)

- a) $47 \times 73 = 3\,431$ et $3\,458 - 3\,431 = 27$, avec $0 \leq 27 < 47 : 3\,458 = 47 \times 73 + 27$, quotient 73 et reste 27. (1)
- b) L'entier vaut $26 \times 31 + 19 = 806 + 19 = 825$; la condition $0 \leq 19 < 26$ est bien vérifiée. (1)
- c) $4n + 5 = 4(n - 2) + 13$, donc $n - 2$ divise $4n + 5$ si et seulement si $n - 2$ divise 13. Les diviseurs de 13 sont ± 1 et ± 13 , et $n - 2 \neq 0$: pour n entier naturel, $n - 2 \in \{-1 ; 1 ; 13\}$, d'où $n = 1$, $n = 3$ ou $n = 15$. (1)
- d) Trois entiers consécutifs s'écrivent $n - 1$, n et $n + 1$; leur somme vaut $(n - 1) + n + (n + 1) = 3n$, qui est un multiple de 3. (1)

Erreur fréquente — Un reste doit toujours vérifier $0 \leq r < b$: si le reste dépasse le diviseur, c'est que le quotient a été sous-estimé.

Exercice 2 – PGCD et fractions

(4 pts)

- a) $2\,431 = 1 \times 1\,683 + 748$; $1\,683 = 2 \times 748 + 187$; $748 = 4 \times 187 + 0$. Le dernier reste non nul est 187, donc $\text{PGCD}(2\,431 ; 1\,683) = 187$. (1)
- b) $1\,683 : 187 = 9$ et $2\,431 : 187 = 13$, donc la fraction s'écrit $9/13$. (1)
- c) $9 = 3^2$ et 13 est premier et ne divise pas 9 : aucun facteur premier commun, donc $\text{PGCD}(9 ; 13) = 1$. (1)
- d) Tout diviseur commun d divise $(n + 8) - (n + 5) = 3$, donc d vaut 1 ou 3. Si $n \equiv 1 [3]$, alors $n + 5 \equiv 0 [3]$ et $n + 8 \equiv 0 [3]$: le PGCD vaut 3. Sinon, 3 ne divise pas $n + 5$ et le PGCD vaut 1. (1)

Le point clé — Le PGCD est le dernier reste non nul de l'algorithme, jamais le reste nul qui met fin aux divisions.

Exercice 3 – Nombres premiers et décomposition

(4 pts)

- a) $35^2 = 1\,225 \leq 1\,271$ et $36^2 = 1\,296 > 1\,271$: on teste les premiers jusqu'à 35. On trouve $1\,271 = 31 \times 41$, donc 1 271 n'est pas premier. (1)
- b) $6\,084 = 2 \times 3\,042 = 2^2 \times 1\,521$; $1\,521 = 3 \times 507 = 3^2 \times 169 = 3^2 \times 13^2$. Donc $6\,084 = 2^2 \times 3^2 \times 13^2$, et le nombre de diviseurs vaut $(2 + 1)(2 + 1)(2 + 1) = 27$. (1)
- c) Tous les exposants sont pairs : $6\,084 = (2 \times 3 \times 13)^2 = 78^2$, donc 6 084 est un carré parfait de racine 78. (1)
- d) $1\,271 = 31 \times 41$ et $6\,084 = 2^2 \times 3^2 \times 13^2$: les facteurs premiers 31 et 41 n'apparaissent pas dans la seconde décomposition, donc aucun premier ne divise les deux entiers et $\text{PGCD}(1\,271 ; 6\,084) = 1$. (1)

Repère — Un entier est un carré parfait si et seulement si tous les exposants de sa décomposition en facteurs premiers sont pairs.

Exercice 4 – Congruences*(4 pts)*

- a) $8 \equiv -1 [9]$, donc $8^{17} \equiv (-1)^{17} \equiv -1 \equiv 8 [9]$. Comme $0 \leq 8 < 9$, le reste vaut 8. (1)
- b) $7 = 6 + 1$, donc $7 \equiv 1 [6]$ et $7^n \equiv 1^n \equiv 1 [6]$. Ainsi $7^n - 1 \equiv 0 [6]$: 6 divise $7^n - 1$. (1)
- c) Modulo 10 : $3^1 \equiv 3$, $3^2 \equiv 9$, $3^3 \equiv 7$, $3^4 \equiv 1$, le cycle a pour longueur 4. Or $101 = 4 \times 25 + 1$, donc $3^{101} \equiv 3 [10]$: le chiffre des unités est 3. (1)
- d) $4 \times 2 = 8 \equiv 1 [7]$, donc en multipliant la congruence par 2 : $8n \equiv 6 [7]$, soit $n \equiv 6 [7]$. Les solutions sont les entiers $n = 6 + 7k$ avec k entier naturel : 6, 13, 20, 27, ... (1)

À retenir — Remplacer la base par -1 ou par 1 modulo n transforme une puissance monstrueuse en un calcul de deux lignes.

Exercice 5 – Algorithmique en Python*(4 pts)*

- a) La fonction calcule le reste de a^n par m : $6^5 = 7\,776$ et $7\,776 = 13 \times 598 + 2$, donc l'appel renvoie 2. (1)
- b) Sans réduction intermédiaire, la variable r atteindrait la taille de a^n , soit des milliers de chiffres pour un grand exposant ; en réduisant à chaque tour, r reste toujours inférieur à m , et le résultat est le même car la congruence est compatible avec le produit. (1)
- c) `def divise(a, b) : renvoyer b % a == 0` — en Python : « `return b % a == 0` » ; la comparaison produit directement un booléen, sans `if ni else`. (1)
- d) L'instruction `b % a` effectue une division par a : pour $a = 0$, Python lève une erreur de division par zéro (`ZeroDivisionError`). Mathématiquement, 0 ne divise que 0. On corrige en traitant ce cas à part : si $a = 0$, renvoyer `True` lorsque $b = 0$ et `False` sinon. (1)

Attention — Le cas du diviseur nul n'est pas un détail d'informatique : 0 ne divise que 0, et tout programme d'arithmétique doit le prévoir.