

BROUILLON - CARRÉS PARFAITS ET PRODUITS D'ENTRIERS CONSÉCUTIFS – JUSQU'À 100 FACTEURS ?

CHRISTOPHE BAL

*Document, avec son source L^AT_EX, disponible sur la page
<https://github.com/bc-writing/drafts>.*

Mentions « légales »

Ce document est mis à disposition selon les termes de la licence Creative Commons « Attribution – Pas d'utilisation commerciale – Partage dans les mêmes conditions 4.0 International ».



TABLE DES MATIÈRES

1.	Ce qui nous intéresse	2
2.	Notations utilisées	2
3.	Les carrés parfaits	2
3.1.	Structure	2
3.2.	Distance entre deux carrés parfaits	2
4.	Une démonstration intéressante	4
5.	Une tactique informatique	5
5.1.	Deux algorithmes basiques	5
5.1.1.	Sélection de potentiels bons candidats	5
5.1.2.	Cibler la recherche pour des tests brutaux	8
5.2.	Les cas gagnants	9
5.3.	Que faire des cas perdants ?	9
6.	Conclusion	9
7.	Sources utilisées	9
8.	AFFAIRE À SUIVRE...	10

1. CE QUI NOUS INTÉRESSE

Dans l'article « *Note on Products of Consecutive Integers* »¹, Paul Erdős démontre que pour tout couple $(n, k) \in \mathbb{N}^* \times \mathbb{N}^*$, le produit de $(k+1)$ entiers consécutifs $n(n+1) \cdots (n+k)$ n'est jamais le carré d'un entier. Plus précisément, l'argument général de Paul Erdős est valable pour $k+1 \geq 100$, soit à partir de 100 facteurs.

Dans ce document, via l'outil informatique principalement, nous nous proposons de traiter les cas laissés de côté par Paul Erdős.

2. NOTATIONS UTILISÉES

Dans la suite, nous emploierons les notations suivantes.

- $\forall (n, k) \in (\mathbb{N}^*)^2, \pi_n^k = \prod_{i=0}^{k-1} (n+i)$.

Par exemple, $\pi_n^1 = n$, $\pi_n^2 = n(n+1)$ et $\pi_{n+2}^4 = (n+2)(n+3)(n+4)(n+5)$.

- ${}^2\mathbb{N} = \{n^2, n \in \mathbb{N}\}$ est l'ensemble des carrés parfaits. On note aussi ${}^2_*\mathbb{N} = {}^2\mathbb{N} \cap \mathbb{N}^*$.
 \mathbb{N}_{sf} est l'ensemble des naturels non nuls sans facteur carré².

- \mathbb{P} désigne l'ensemble des nombres premiers.

$\forall (p; n) \in \mathbb{P} \times \mathbb{N}^*$, $v_p(n) \in \mathbb{N}$ est la valuation p -adique de n , c'est-à-dire $p^{v_p(n)} \mid n$ et $p^{v_p(n)+1} \nmid n$, autrement dit $p^{v_p(n)}$ divise n , contrairement à $p^{v_p(n)+1}$.

- $2\mathbb{N}$ désigne l'ensemble des nombres naturels pairs.

$2\mathbb{N} + 1$ est l'ensemble des nombres naturels impairs.

- $\text{reste}(a, b)$ désigne le reste de la division euclidienne de $a \in \mathbb{N}$ par $b \in \mathbb{N}^*$.
 $\text{quot}(a, b)$ désigne le quotient de la division euclidienne de $a \in \mathbb{N}$ par $b \in \mathbb{N}^*$.
- $[x]$ désigne la partie entière de $x \in \mathbb{R}$.

3. LES CARRÉS PARFAITS

3.1. Structure.

Fait 3.1. $n \in {}^2_*\mathbb{N}$ si, et seulement si, $\forall p \in \mathbb{P}, v_p(n) \in 2\mathbb{N}$.

Démonstration. Immédiat à valider. □

3.2. Distance entre deux carrés parfaits.

Fait 3.2. Soit $(N, M) \in \mathbb{N}^* \times \mathbb{N}^*$ tel que $N > M$.

(1) $N^2 - M^2 \geq 2N - 1$.

(2) Notons nb_{sol} le nombre de solutions $(N, M) \in \mathbb{N}^* \times \mathbb{N}^*$ de $N^2 - M^2 = \delta$.

Pour $\delta \in \llbracket 1; 10 \rrbracket$, nous avons :

(a) $nb_{sol} = 0$ si $\delta \in \{1, 2, 4, 6, 10\}$.

(b) $nb_{sol} = 1$ si $\delta \in \{3, 5, 7, 8, 9\}$. Ainsi, $N^2 - M^2 = 3$ uniquement si $(M, N) = (1, 2)$.

1. J. London Math. Soc. 14 (1939).

2. En anglais, on dit « square free ».

Démonstration.

- (1) Comme $N - 1 \geq M$, nous obtenons : $N^2 - M^2 \geq N^2 - (N - 1)^2 = 2N - 1$.
- (2) Le point précédent permet d'utiliser le programme Python suivant afin d'obtenir rapidement les listes de nombres indiquées. \square

```
from collections import defaultdict
from math          import sqrt, floor

# N**2 - M**2 = diff ?
def sol(diff):
    solfound = []

    for N in range(1, (diff + 1) // 2 + 1):
        M_square = N**2 - diff

        if M_square > 0:
            M = floor(sqrt(M_square))

            if M != 0 and M**2 == M_square:
                solfound.append((N, M))

    return solfound

all_nbsol = defaultdict(list)

for d in range(1, 101):
    all_nbsol[len(sol(d))].append(d)

print(all_nbsol)
```

Remarque 3.1. La fonction `sol` du programme précédent traduit l'algorithme suivant que nous utiliserons dans la suite de ce document.

Algorithme 1

Donnée : $\delta \in \mathbb{N}^*$

Résultat : \mathcal{S} l'ensemble des couples $(N, M) \in \mathbb{N}^* \times \mathbb{N}^*$ vérifiant $N^2 - M^2 = \delta$.

Actions

$\mathcal{S} \leftarrow \emptyset$

Pour N allant de 1 jusqu'à $1 + \text{quot}(\delta, 2)$:

$M_{\text{square}} \leftarrow N^2 - \delta$

Si $M_{\text{square}} > 0$:

$M \leftarrow \lfloor \sqrt{M_{\text{square}}} \rfloor$

Si $M \neq 0$ et $M^2 = M_{\text{square}}$:

$\mathcal{S} \leftarrow \mathcal{S} \cup \{(N, M)\}$

Renvoyer \mathcal{S}

4. UNE DÉMONSTRATION INTÉRESSANTE

Nous allons présenter dans cette section une démonstration qui va nous donner sans effort un algorithme permettant de valider, ou rejeter, la proposition $\pi_n^k \notin {}^2\mathbb{N}$ au cas par cas³.

Dans un échange sur <https://math.stackexchange.com>, voir la section 7, il est indiqué une preuve de $\pi_n^{10} \notin {}^2\mathbb{N}$ pour $n \in \mathbb{N}^*$ quelconque. Voici cette preuve complétée avec certains arguments laissés sous silence dans la source utilisée. Noter au passage que l'essentiel consiste en des actes algorithmiques basiques permettant d'appliquer le fait 3.2 sans user de fourberies déductives. Ennuyeux, mais efficace !

Preuve. Supposons que $\pi_n^{10} \in {}^2\mathbb{N}$ pour $n \in \mathbb{N}^*$.

Clairement, $\forall p \in \mathbb{P}_{\geq 10}, \forall i \in \llbracket 0; 9 \rrbracket, v_p(n+i) \in 2\mathbb{N}$. Concentrons-nous sur les nombres premiers dans $\{2, 3, 5, 7\}$. Voici ce que l'on peut observer très grossièrement.

- Au maximum deux facteurs $(n+i)$ de π_n^{10} sont divisibles par 5.
- Au maximum deux facteurs $(n+i)$ de π_n^{10} sont divisibles par 7.
- Les points précédents donnent au moins 6 facteurs $(n+i)$ de π_n^{10} de valuation p -adique paire dès que $p \in \mathbb{P}_{\geq 5}$.

Nous avons alors l'une des alternatives suivantes pour chacun des 6 facteurs $(n+i)$ vérifiant $v_p(n+i) \in 2\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq 5}$.

- **[A1]** $(v_2(n+i), v_3(n+i)) \in 2\mathbb{N} \times 2\mathbb{N}$
- **[A2]** $(v_2(n+i), v_3(n+i)) \in 2\mathbb{N} \times (2\mathbb{N} + 1)$
- **[A3]** $(v_2(n+i), v_3(n+i)) \in (2\mathbb{N} + 1) \times 2\mathbb{N}$
- **[A4]** $(v_2(n+i), v_3(n+i)) \in (2\mathbb{N} + 1) \times (2\mathbb{N} + 1)$

Comme nous avons six facteurs pour quatre alternatives, ce bon vieux principe des tiroirs va nous permettre de lever des contradictions.

- Deux facteurs différents $(n+i)$ et $(n+i')$ vérifient **[A1]**.

Dans ce cas, $(n+i, n+i') = (N^2, M^2)$ avec $(M, N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $N^2 - M^2 \in \llbracket 1; 9 \rrbracket$. Selon le fait 3.2, seuls les cas suivants sont possibles mais ils lèvent tous une contradiction.

- (1) $N^2 - M^2 = 3$ avec $(M, N) = (1, 2)$ est possible, mais ceci donne $n = 1^2 = 1$, puis $\pi_1^{10} = 10! \in {}^2\mathbb{N}$, or ceci est faux car $v_7(10!) = 1$.
- (2) $N^2 - M^2 = 5$ avec $(M, N) = (2, 3)$ est possible d'où $n \in \llbracket 1; 4 \rrbracket$. Nous venons de voir que $n = 1$ est impossible. De plus, pour $n \in \llbracket 2; 4 \rrbracket$, $v_7(\pi_n^{10}) = 1$ montre que $\pi_n^{10} \in {}^2\mathbb{N}$ est faux.
- (3) $N^2 - M^2 = 7$ avec $(M, N) = (3, 4)$ est possible d'où $n \in \llbracket 1; 9 \rrbracket$, puis $n \in \llbracket 5; 9 \rrbracket$ d'après ce qui précède. Mais ici, $\forall n \in \llbracket 5; 9 \rrbracket, v_{11}(\pi_n^{10}) = 1$ montre que $\pi_n^{10} \in {}^2\mathbb{N}$ est faux.
- (4) $N^2 - M^2 = 8$ avec $(M, N) = (1, 3)$ est possible d'où $n = 1$, mais ceci est impossible comme nous l'avons vu ci-dessus.
- (5) $N^2 - M^2 = 9$ avec $(M, N) = (4, 5)$ est possible d'où $n \in \llbracket 10; 16 \rrbracket$ d'après ce qui précède. Or $\forall n \in \llbracket 10; 16 \rrbracket, v_{17}(\pi_n^{10}) = 1$, donc $\pi_n^{10} \in {}^2\mathbb{N}$ est faux.

3. Ou au k par k ...

- Deux facteurs différents $(n + i)$ et $(n + i')$ vérifient [A2].

Dans ce cas, $(n + i, n + i') = (3N^2, 3M^2)$ avec $(M, N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $3(N^2 - M^2) \in \llbracket 1; 9 \rrbracket$, puis $N^2 - M^2 \in \llbracket 1; 3 \rrbracket$. Selon le fait 3.2, nécessairement $N^2 - M^2 = 3$ avec $(M, N) = (1, 2)$, d'où $n \in \llbracket 1; 3 \rrbracket$, mais on sait que cela est impossible.

- Deux facteurs différents $(n + i)$ et $(n + i')$ vérifient [A3].

Dans ce cas, $(n + i, n + i') = (2N^2, 2M^2)$ avec $(M, N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $2(N^2 - M^2) \in \llbracket 1; 9 \rrbracket$, puis $N^2 - M^2 \in \llbracket 1; 4 \rrbracket$. Selon le fait 3.2, nécessairement $N^2 - M^2 = 3$ avec $(M, N) = (1, 2)$, d'où $n \in \llbracket 1; 2 \rrbracket$, mais on sait que cela est impossible.

- Deux facteurs différents $(n + i)$ et $(n + i')$ vérifient [A4].

Dans ce cas, $(n + i, n + i') = (6N^2, 6M^2)$ avec $(M, N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $6(N^2 - M^2) \in \llbracket 1; 9 \rrbracket$, puis $N^2 - M^2 = 1$, mais c'est impossible d'après le fait 3.2. \square

Ce qui est intéressant avec la preuve précédente est qu'avec quelques adaptations « mécaniques », on démontre que $\forall n \in \mathbb{N}^*, \pi_n^k \notin {}^2\mathbb{N}$ dès que $k \in \{3, 5, 7, 9, 11, 12, 13\}$.⁴ Ces preuves semblant peu gourmandes informatiquement, il semble opportun de tenter un traitement numérique des cas absents dans l'article de Paul Erdős.

5. UNE TACTIQUE INFORMATIQUE

5.1. Deux algorithmes basiques.

La démonstration donnée dans la section 4 part de $\pi_n^k \in {}^2\mathbb{N}$ par hypothèse, puis elle s'appuie sur deux idées simples que nous allons transformer en algorithme.

5.1.1. Sélection de potentiels bons candidats.

La première phase consiste à tenter de trouver le moins possible de nombres premiers p tel que tous les facteurs $(n + i)$ de π_n^k soient de valuation p -adique non nécessairement paire. Comme $p \in \mathbb{P}_{\geq k}$ divise au maximum un facteur $(n + i)$ de π_n^k , nous avons $\forall i \in \llbracket 0; k - 1 \rrbracket$, $v_p(n + i) \in 2\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq k}$ puisque $\pi_n^k \in {}^2\mathbb{N}$ par hypothèse. Ceci permet de cibler notre analyse sur les nombres premiers dans $\mathbb{P}_{<k}$.

Voici un premier exemple de sélection avec π_n^3 en notant que $\mathbb{P}_{<3} = \{2\}$. Nous expliquons juste après comment lire le tableau ci-dessous.

p_m		2
Occu. max.		2
Occu. libres.	3	1
Alternatives.	2^1	0

Le tableau se lit comme suit.

- p_m désigne le plus grand nombre premier disponible non encore éliminé.
- La deuxième ligne indique le nombre maximum de facteurs $(n + i)$ de π_n^k pouvant être divisibles par p_m .

4. Voir mon document « Carrés parfaits et produits d'entiers consécutifs – Des solutions à la main » disponible dans le dépôt associé au présent document.

- La troisième ligne donne le nombre minimum de facteurs de valuations p -adiques nécessairement paires dès que $p \in \mathbb{P}_{\geq p_m}$.
- La dernière ligne donne le nombre d'alternatives possibles relativement aux parités des valuations p -adiques pour les nombres premiers p dans $\mathbb{P}_{< p_m}$, les autres valuations p -adiques restantes étant paires.
- La colonne sur fond vert indique le « meilleur bon » candidat, c'est-à-dire celui avec le moins d'alternatives. Nous utiliserons du bleu pour de bons candidats non gardés.
- La colonne sur fond rouge indique que l'on ne peut plus avancer (évident ici mais nous verrons que cela peut arriver plus tôt dans l'analyse).

Nous voyons ici que 2 est un bon candidat pour rejeter $\pi_n^3 \in {}^2\mathbb{N}$ puisqu'au moins deux facteurs différents $(n+i)$ et $(n+i')$ de π_n^3 vérifient la même alternative, d'où $n+i = cM^2$ et $n+i' = cN^2$ avec $(c, N, M) \in \mathbb{N}_{sf} \times (\mathbb{N}^*)^2$, une information qui sera utilisée par notre second algorithme pour « localiser », via le fait 3.2, des entiers naturels n afin de tester presque brutalement si $\pi_n^3 \in {}^2\mathbb{N}$ est vrai, ou non.

Voici un autre exemple montrant que la sélection peut échouer : il suffit de considérer par exemple π_n^4 en notant que $\mathbb{P}_{<4} = \{2, 3\}$.

p_m		3	2
Occu. max.		2	2
Occu. libres.	4	2	0
Alternatives.	2^2	2^1	0

Afin de clarifier la démarche que nous allons suivre, donnons un dernier exemple via π_n^{37} en notant que $\text{card}(\mathbb{P}_{<37}) = 11$.

p_m		31	29	23	19	17	13	11	7	5	3
Occu. max.		2	2	2	2	3	3	4	6	8	13
Occu. libres.	37	35	33	31	29	26	23	19	13	5	0
Alternatives.	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2

Nous décidons donc de procéder grosso modo comme suit.

- (1) Nous supposons par l'absurde que $\pi_n^k \in {}^2\mathbb{N}$ avec $k \in \mathbb{N}^*$.
- (2) Nous fabriquons $\mathcal{P} = \mathbb{P}_{<k}$.
- (3) Nous posons $\mathcal{C} = \emptyset$.

Cet ensemble sera celui des nombres premiers « candidats » utilisés dans notre second algorithme de tests brutaux. Nous cherchons à obtenir l'ensemble \mathcal{C} non vide le plus petit possible.

- (4) Nous posons aussi $\text{occu}_{\text{libre}} = k$.

Cette variable va nous servir à compter les facteurs $(n+i)$ de π_n^k ayant un « maximum » de valuations p -adiques paires.

- (5) **Début des actions répétitives.**

Si $\mathcal{P} = \emptyset$, ou $\text{occu}_{\text{libre}} = 0$, nous stoppons tout !

Si $\mathcal{C} = \emptyset$, nous avons perdu. Dans le cas contraire, nous pourrions continuer avec l'algorithme qui sera présenté dans la section 3 suivante.

- (6) Si $occu_{libre} > 2^{\text{card}(\mathcal{P})}$, nous posons $\mathcal{C} = \mathcal{P}$.

Nous avons $2^{\text{card}(\mathcal{P})}$ alternatives $[A_j]$ possibles relativement aux parités possible des valuations p -adiques pour les nombres premiers p dans $\mathcal{P} = \mathbb{P}_{<p_m}$, les valuations p -adiques restantes étant paires. De l'autre côté, nous avons au moins $occu_{libre}$ facteurs $(n+i)$ de π_n^k tels que $v_p(n+i) \in 2\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq p_m}$. Finalement, si $occu_{libre} > 2^{\text{card}(\mathcal{P})}$, nous avons au moins deux facteurs différents $(n+i)$ et $(n+i')$ vérifiant la même alternative $[A_j]$, d'où $n+i = cM^2$ et $n+i' = cN^2$ avec $(c, N, M) \in \mathbb{N}_{sf} \times (\mathbb{N}^*)^2$, une information qui sera utilisée par notre second algorithme pour « localiser » des π_n^k à tester brutalement.

- (7) Ensuite, nous considérons $p_m = \max(\mathcal{P})$, puis retirons p_m de \mathcal{P} , d'où $\mathcal{P} = \mathbb{P}_{<p_m}$.
Le choix du maximum tente de limiter les rejets de facteurs dans les étapes suivantes.

- (8) Nous calculons $occu_{max}$ le nombre maximum de facteurs $(n+i)$ de π_n^k pouvant être divisés par p_m .

Le calcul de $occu_{max}$ est simple puisqu'il suffit de considérer le cas où p_m divise n , nous avons alors $occu_{max} = 1 + \text{quot}(k-1, p_m)$ car $\pi_n^k = n(n+1) \cdots (n+k-1)$.

- (9) $occu_{libre}$ devient $occu_{libre} - occu_{max}$.

Maintenant, nous savons qu'au moins $occu_{libre}$ facteurs $(n+i)$ de π_n^k vérifient $v_p(n+i) \in 2\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq p_m}$.

- (10) Nous reprenons les étapes à partir du point 5.

Tout ce qui précède nous amène à l'algorithme suivant.

Algorithme 2

Donnée : $k \in \mathbb{N}_{\geq 2}$, le nombre de facteurs considérés.

Résultat : \mathcal{C} un ensemble, éventuellement vide, de nombres premiers « candidats » tel que si $\mathcal{C} \neq \emptyset$ alors il existe au moins deux facteurs $(n+i)$ et $(n+i')$ de π_n^k vérifiant $\forall p \in \mathbb{P} - \mathcal{C}$, $(v_p(n+i), v_p(n+i')) \in (2\mathbb{N})^2$, ainsi que $v_p(n+i)$ et $v_p(n+i')$ ont la même parité dès que $p \in \mathcal{C}$.

Actions

$\mathcal{P} \leftarrow \mathbb{P} \cap \llbracket 0; k-1 \rrbracket$

$\mathcal{C} \leftarrow \emptyset$

$occu_{libre} \leftarrow k$

Tant Que $\mathcal{P} \neq \emptyset$ **et** $occu_{libre} \neq 0$:

Si $occu_{libre} > 2^{\text{card}(\mathcal{P})}$:

$\mathcal{C} \leftarrow \mathcal{P}$

$p_m \leftarrow \max(\mathcal{P})$

$\mathcal{P} \leftarrow \mathcal{P} - \{p_m\}$

$occu_{max} \leftarrow 1 + \text{quot}(k-1, p_m)$

$occu_{libre} \leftarrow \max(0, occu_{libre} - occu_{max})$

Renvoyer \mathcal{C}

Une fois l'algorithme 2 traduit en Python, nous obtenons instantanément les informations suivantes pour $k \in \llbracket 2; 100 \rrbracket$.

- **Mauvais candidats.**

Il y en a 4 qui sont 2, 4, 6 et 8.

- **Bons candidats avec un seul nombre premier à gérer.**

Il y en a 2 qui sont 3 et 5.

- **Bons candidats avec deux nombres premiers à gérer.**

Il y en a 27 qui sont 7, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 25, 26, 27, 28, 29, 30, 31, 33, 34, 35 et 37.

- **Bons candidats avec trois nombres premiers à gérer.**

Il y en a 66 qui sont les entiers restants.

Ce qui précède est encourageant, car peu de cas sont rejetés. De plus, les mauvais candidats sont faciles à gérer par un humain, ou un programme : voir la section 5.3. Quant aux candidats acceptés, les nombres premiers à gérer sont forcément dans $\{2, 3, 5\}$, et le nombre maximum d'alternatives est $2^3 = 8$, tout ceci n'étant pas informatiquement bloquant (nous verrons dans la section 5.1.2 qu'un autre paramètre peut bloquer la recherche).

Remarque 5.1. *Ne rêvons pas trop, car le programme donne aussi que 824 est le premier naturel, après 8, non sélectionné par notre algorithme.*

5.1.2. Cibler la recherche pour des tests brutaux.

Reprenons les notations de la section 5.1.1 en supposant $\pi_n^k \in {}^2_*\mathbb{N}$ avec $k \in \mathbb{N}_{\geq 2}$, et plaçons-nous dans la situation où l'ensemble \mathcal{C} renvoyé par l'algorithme 2 est non vide (dans le cas contraire, notre tactique est mise en échec). Nous avons donc au moins deux facteurs $(n + i)$ et $(n + i')$ de π_n^k vérifiant les points suivants.

- $\forall p \in \mathbb{P} - \mathcal{C}, (v_p(n + i), v_p(n + i')) \in (2\mathbb{N})^2$.
- $\forall p \in \mathcal{C}, v_p(n + i)$ et $v_p(n + i')$ ont la même parité.

Ceci permet de « localiser » comme suit les valeurs de n pouvant vérifier $\pi_n^k \in {}^2_*\mathbb{N}$.

(1) Notons \mathcal{C} l'ensemble des entiers $\prod_{p \in \mathcal{C}} p^{(\epsilon_p)}$ avec $(\epsilon_p)_{p \in \mathcal{C}} \subseteq \{0, 1\}$.

(2) Il existe $c \in \mathcal{C}$ tel que $n + i = cM^2$ et $n + i' = cN^2$ avec $(c, M, N) \in \mathbb{N}_{sf} \times (\mathbb{N}^*)^2$.

Comme nous n'avons aucune idée de la suite $(\epsilon_p)_{p \in \mathcal{C}}$ à choisir, il faudra les tester toutes.

(3) $c(N^2 - M^2) \in \llbracket 1; k - 1 \rrbracket$ donne $N^2 - M^2 \in \llbracket 1; \text{quot}(k - 1, c) \rrbracket$.

(4) L'algorithme 1 permet de construire l'ensemble fini \mathcal{D}_c , éventuellement vide, des couples (M, N) tels que $N^2 - M^2 \in \llbracket 1; \text{quot}(k - 1, c) \rrbracket$, à partir duquel nous construisons $\mathcal{F}_c = \{(cM^2, cN^2) \mid (M, N) \in \mathcal{D}_c\}$.

Nous pouvons affirmer que π_n^k contient au moins un couple de facteurs (f, f') appartenant à $\cup_{c \in \mathcal{C}} \mathcal{F}_c$, et vérifiant $f < f'$.

(5) XXX

(6) XXX

(7) XXX

(8) XXX

???

Tout ce qui précède nous amène à l'algorithme suivant.

Algorithme 3

Donnée : ?

Résultat : ?

Actions

⌊ ?

5.2. Les cas gagnants.

Une fois les algorithmes 1, 2 et 3 traduits en Python⁵, nous validons sans effort que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in \llbracket 2; 100 \rrbracket - \{2, 4, 6, 8\}$.

5.3. Que faire des cas perdants ?

Aussi surprenant que cela puisse paraître, il est très facile, bien que fastidieux, de démontrer humainement que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in \{2, 4, 6, 8\}$: voir mon document « *Carrés parfaits et produits d'entiers consécutifs – Une méthode efficace* » pour savoir comment cela fonctionne. La méthode présentée étant facile à coder, un programme Python, fait sans astuce, démontre instantanément, ou presque, que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in \llbracket 2; 8 \rrbracket$.

6. CONCLUSION

Nous avons démontré informatiquement que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in \llbracket 2; 100 \rrbracket$. Il ne reste plus qu'à lire, et comprendre pleinement, l'article « *Note on Products of Consecutive Integers* » de Paul Erdős. Bon courage !

7. SOURCES UTILISÉES

L'idée centrale de ce document vient d'une source citée dans un échange consulté le 13 février 2024, et titré « *Product of 10 consecutive integers can never be a perfect square* » sur le site <https://math.stackexchange.com>.

5. Voir sur le dépôt associé à ce document.

8. AFFAIRE À SUIVRE...
