Trois exercices indépendants

Exercice 1

Dans le plan rapporté au repère (O, u, v), on considère la "grille" des points $M(x, y)$ donc les cooronnées x, y sont des entiers naturels.
On se donne un entier naturel n.
Un pion placé initialement en $(0, 0)$ suit une trajectoire de n mouvements successifs sur la grille.
A chaque étape, seulement trois mouvements sont autorisés :

— D'une position "vers l'est" (translation de vecteur u, de (x, y) à $(x+1, y)$)

— D'une position "vers le nord" (translation de vecteur v, de (x, y) à $(x, y+1)$)

— D'une position "vers le nord-est" (translation de vecteur $u+v$, de (x, y) à $(x+1, y+1)$)

1. Quel est le nombre total de trajectoires possibles ?
2. On note A_n des points que peut atteindre le pion à l'issue de ses n déplacements.
   Montrer que A_n est l'ensemble des points $M(x, y)$, avec $0 \leqslant x \leqslant n$, $0 \leqslant y \leqslant n$, et $x + y \geqslant n$.
3. Soit $M(x, y)$ un point de l'ensemble A_n.
   Montrer que le nombre de trajectoires qui aboutissent à $M(x, y)$ est $\binom{n}{x}\binom{x}{n-y}$.
4. Dans cette question, on suppose $n = 10$.
   On suppose qu'à chaque étape, les trois déplacements possibles sont équiprobables.
   Avec votre calculatrice, calculer de façon exacte puis approchée la probabilité pour que la position finale $M(x, y)$ du pion soit sur la diagonale $y = x$.

Exercice 2

On définit une suite $(a_n)_{n\geqslant 0}$ par $a_0 = 1$ et $\begin{cases} a_{2n+1} = a_n \\ a_{2n+2} = a_n + a_{n+1} \end{cases}$ pour tout $n \geqslant 0$.

1. Soient r et s deux entiers premiers entre eux. Montrer qu'il existe n tel que $a_n = r$ et $a_{n+1} = s$.
   Indication : récurrence forte sur la valeur de $r + s$.
2. Écrire une fonction Python `a` (prenant en argument un entier $n \geqslant 0$, et renvoyant a_n).
   La fonction `a` utilisera bien sûr la définition récursive de la suite $(a_n)_{n\geqslant 0}$.
   On utilisera un *dictionnaire* A (formé d'enregistrements $k : a_k$, et initialisé par $A = \{0 : 1\}$, c'est-à-dire $a_0 = 1$), pour mémoriser les résultats des appels récursifs de a.
3. Écrire quelques lignes de Python pour afficher les 100 premières valeurs de la suite a dans un tableau 10×10 (on recommande ici de charger l'extension Numpy).
4. Écrire une fonction Python nommée `gcd`, et calculant le pgcd de deux entiers.
5. Écrire une fonction Python nommée `find`, prenant en arguments deux entiers naturels r et s, et renvoyant un entier n tel que $a_n = r$ et $a_{n+1} = s$.

Exercice 3

On écrit à la suite tous les entiers de 1 jusqu'à 2013.
On les efface de 3 en 3, en commençant par le premier (on efface donc 1, 4, 7, etc.)
On recommence alors la même opération sur la liste restante, et ainsi de suite.
On note a_n l'entier qui est en tête après n itérations ($a_0 = 1, a_1 = 2, a_2 = 3, a_3 = 5, a_4 = 8, \cdots$)
Exprimer a_{n+1} en fonction de a_n (indication : discuter suivant la parité de a_n.)

1. Combien d'itérations faut-il pour que la liste initiale soit complètement effacée ?
2. Écrire une fonction Python prenant en argument une liste L et un entier n (ce dernier, facultatif, prenant la valeur 1 par défaut), et qui renvoie la liste obtenue à partir de L par n effacements successifs (en respectant la règle spécifiée dans l'énoncé).