Les exercices proposés dans cette feuille peuvent être traités à l'aide d'un tableur de votre choix ou d'un logiciel de programmation de votre choix également.

1. Jeu de dés

On lance deux dés à 6 faces équilibrés.
Si la différence (en valeur absolue) des points obtenus est inférieure ou égale à 2, alors le meneur de jeu gagne, sinon il perd.

Simuler plusieurs parties... Le jeu semble-t-il équilibré ?

2. Suites à l'épreuve pratique

On considère la suite récurrente (u_n) de premier terme $u_0 = 0$ et telle que pour tout entier naturel n , $u_{n+1} = u_n + 2n - 11$.

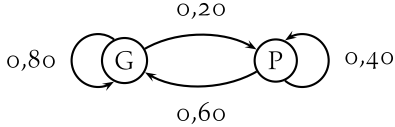
1. Calculer et représenter graphiquement les 20 premiers termes de cette suite.
2. Étant donné n, on peut calculer la valeur de u_n si on connaît la valeur de u_{n+1}. On voudrait à présent pouvoir calculer, pour n'importe quelle valeur de l'entier naturel non nul n, la valeur de u_n sans pour autant connaître la valeur de u_{n-1}. Pour cela il faudrait disposer d'une formule donnant u_n en fonction de n.
   a) Conjecturer cette formule.
   b) Démontrer cette formule.

Conseil : Utiliser une *courbe de tendance* et/ou un logiciel de calcul formel.

3. Les puces

Marius est un dresseur de puces qui a un excellent numéro de cirque : il possède 1 000 puces qu'il fait sauter en l'air en cadence.
Il place deux podiums l'un à côté de l'autre : un petit (P) et un grand (G), et il a constaté que :

- parmi les puces placées sur le grand podium (G), 80 % retombent sur place et 20 % sur le petit podium ;
- parmi les puces placées sur le petit podium (P), 40 % retombent sur place et 60 % sur le grand podium.



Il fait sauter les puces plusieurs fois de suite.
On suppose qu'il y a initialement 200 puces sur le grand podium et 800 sur le petit.
On souhaite savoir comment évolue la répartition des puces sur les podiums, et notamment savoir si celle-ci semble se stabiliser.

Pour ouvrir le problème

- le nombre de puces par podium doit être un nombre entier !
- démontrer avec la matrice associée au graphe.

4. Syracuse

La suite de Syracuse est définie par $u_0 \in \mathbb{N}$ et $u_{n+1} = \begin{cases} \dfrac{u_n}{2} & \text{si } u_n \text{ est pair} \\ 3u_n + 1 & \text{sinon} \end{cases}$

On définit :

le temps de vol : c'est le plus petit indice $n > 0$ tel que $u_n = 1$.

le temps de vol en altitude : c'est le plus petit indice n tel que $u_{n+1} < u_0$.
l'altitude maximale : c'est la valeur maximale de la suite

Dessiner une ligne brisée joignant les premiers termes de la suite. u_0 doit pouvoir varier et le graphique est alors actualisé.
À l'aide d'un logiciel permettant de programmer : donner l'altitude maximale en fonction de u_0 .

Vous pourrez utiliser l'algorithme suivant :

Données :	la valeur de u_0
Sortie :	la valeur de h représentant l'altitude maximale
Traitement :	h prend la valeur de u_0
	u prend la valeur de u_0
	Tant que $u > 1$ faire
	si u est pair alors
	u prend la valeur $\frac{u}{2}$
	sinon
	u prend la valeur $3 \times u + 1$
	fin si
	h prend la valeur $\max(h, u)$
	Fintantque