

Exercice 1

- 1) Déterminer la somme des multiples de 11 inférieurs à 1000
- 2) Calculer la somme $4 + 7 + 10 + \dots + 64$
- 3) Calculer la somme $1 + 0,5 + 0,5^2 + \dots + 0,5^{12}$
- 4) (u_n) est une suite géométrique de raison $\frac{4}{5}$ et de 1^{er} terme 10. Calculer la somme des 10 premiers termes de la suite.

Exercice 2

Au mois de janvier, un constructeur automobile a construit 155 voitures. Sa production augmente de 6 unités par mois tout au long de l'année.

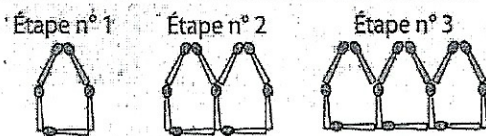
- 1) Combien de voitures ont été fabriquées au mois de septembre ?
- 2) A quoi sert l'algorithme ci-dessous ?

```
def nbvoitures():
    u=155
    S=u
    for i in range (2,13):
        u=u+6
        S=S+u
    return S
```

- 3) Calculer le nombre total de voitures fabriquées par ce constructeur sur une année.

Exercice 3

On schématise avec des allumettes des maisons dont les murs sont mitoyens, comme ci-dessous :



- 1) Combien d'allumettes sont utilisées à l'étape n° 2020 ?
- 2) On note u_n le nombre d'allumettes à l'étape n. Exprimer u_n en fonction de n .
- 3) Déterminer le nombre d'allumettes nécessaires au total pour construire toutes les étapes de l'étape 1 à l'étape 15
- 4) Compléter l'algorithme suivant permettant de déterminer le nombre d'allumettes utilisées si on construit toutes les étapes de l'étape 1 à l'étape n

```
def allumettes(n):
    u=...
    S=u
    for i in range (2,n+1):
        u=...
        S=...
    return ...
```

Exercice 4

Sofia se place près d'un lac et décide de s'entraîner à faire des ricochets. Elle lance un caillou qui parcourt 2 m avant de rebondir une 1^{ère} fois. Le caillou parcourt ensuite la moitié, donc 1 m, avant de rebondir une 2^{ème} fois. Puis il parcourt la moitié, donc 0,5 m, avant de rebondir une 3^{ème} fois. Le caillou continue indéfiniment comme cela.

- 1) Quelle est la distance parcourue par le caillou au moment du 20^e rebond ?
- 2) Quelle est la distance parcourue par le caillou au moment du n -ième rebond ?
- 3) Pour un très grand nombre de rebonds, vers quelle valeur se rapproche la distance totale parcourue par le caillou ?

Exercice 5

Une légende raconte que le roi des Indes aurait voulu récompenser l'inventeur du jeu d'échecs, Sissa (V^e siècle), en lui demandant ce qu'il souhaitait.

Celui-ci lui demanda qu'on déposât un grain de riz sur la $1^{ère}$ case de l'échiquier, 2 sur la $2^{ème}$ case, 4 sur la 3^{e} case, 8 sur la 4^{e} case, ...

- 1) Sachant qu'un jeu d'échecs compte 64 cases, quel est le nombre de grains de riz demandé par Sissa ?
- 2) Sachant qu'un grain de riz pèse en moyenne 0,02 g, comparer la masse totale des grains de riz à déposer sur le jeu avec la production mondiale de riz en 2017 (749 millions de tonnes).
Le roi a-t-il pu exaucer le souhait de Sissa ?

Exercice 6

Kenza veut comparer les prix de deux mutuelles entre un assureur A et un assureur B.

Pour chaque assureur, le prix initial proposé est 300 euros (l'année) en 2023.

- A. L'assureur A prévoit une augmentation de 10 € par an. On note u_n le prix annuel de la mutuelle de l'assureur A en $2023 + n$

- 1) Quelle est la nature de la suite (u_n) ? Exprimer u_n en fonction de n
- 2) Quel sera le prix de la mutuelle de l'assureur A en 2030 ?
- 3) Compléter l'algorithme ci-contre permettant de calculer le montant de la mutuelle de l'assureur A l'année $(2023 + n)$
- 4) Combien Kenza aura-t-elle payé au total en 25 ans si elle choisit l'assureur A ?

```
def mutuelleA(n):  
    u=....  
    for i in range (.....):  
        u=....  
    return .....
```

- B. L'assureur B prévoit une augmentation de 2% par an. On note v_n le prix annuel de la mutuelle de l'assureur B en $2023 + n$

- 1) Quelle est la nature de la suite (v_n) ? Exprimer v_n en fonction de n
- 2) Quel sera le prix de la mutuelle de l'assureur B en 2030 ?
- 3) Combien Kenza aura-t-elle payé au total en 25 ans si elle choisit l'assureur B ?
- 4) Compléter l'algorithme ci-contre permettant de déterminer le montant total payé de 2023 à l'année $(2023 + n)$ en choisissant la mutuelle de l'assureur B

```
def mutuelleB(n):  
    u=....  
    S=.....|  
    for i in range (1,n+1):  
        u=....  
        S=....  
    return .....
```

- C. On compare le montant annuel des deux mutuelles

- A) Compléter l'algorithme ci-contre permettant d'obtenir l'année à partir de laquelle la mutuelle B devient pour la $1^{ère}$ fois supérieur à la mutuelle A
- B) Déterminer la réponse à la calculatrice

```
def comparmutuelles():  
    u=....  
    v=....  
    n=0  
    while ..... :  
        u=....  
        v=....  
        n=....  
    return .....
```