

### Prérequis

1. Quelques notions python : Définition d'une fonction, test, boucle, compteur (un plus).
2. Étude d'une suite.

Ce scénario met en relief le **biais d'ancrage**, un biais cognitif qui décrit la tendance des individus à s'appuyer fortement sur la première information qu'ils reçoivent.

---

#### a) Objectifs des séances :

Amener les élèves à exercer leur esprit critique en prenant appui sur l'histoire de racine 2. Motiver les élèves à s'interroger et à mettre en application leurs réflexions. Développer l'esprit critique en les amenant à comparer les résultats de deux méthodes. Réutiliser les notions de suites dans un nouveau contexte. Développer leurs connaissances en python à travers la compréhension et l'application d'un algorithme. Initier les élèves à la modélisation mathématique et dans un sens plus large à la recherche scientifique.

---

#### b) Démarche de l'enseignant :

L'idée est de faire participer les élèves à une réflexion scientifique leur permettant de faire une comparaison détaillée entre deux méthodes différentes d'approximation de la racine carrée de 2. Au cours de ce scénario, on laissera d'abord les élèves construire leur propre réflexion en partageant leurs idées autour d'un débat préalable. Ensuite on leur proposera d'élaborer en groupe une première méthode d'approximation de  $\sqrt{2}$  à l'aide d'outils simples (ciseaux, corde ou tige de 1m) le but étant de les initier à la modélisation mathématique et d'aiguiser leur esprit critique.

---

#### c) Déroulement du scénario :

Dans un premier temps, aucune mise en scène particulière n'est attendue. On expliquera l'objet de l'étude à travers l'histoire de l'approximation de  $\sqrt{2}$  obtenue par les babyloniens et par les égyptiens. On demandera ensuite aux élèves de conduire l'expérience qui consiste à construire un carré de côté 1m à l'aide de cordes puis de retrouver une approximation de la diagonale du carré en utilisant uniquement des cordes de 1m et des ciseaux. On procédera par la suite à l'explication de la méthode de Héron, d'abord à travers un schéma initiant cette méthode puis par l'exécution de l'algorithme à la main et à l'aide d'un outil numérique.

## Méthode de Héron

Le procédé de l'algorithme de Héron est très simple. En prenant un rectangle de côté 1 et 2, l'aire de ce rectangle est égale à 2. La racine carrée de 2 correspond au côté du carré que l'on obtiendrait en déformant notre rectangle mais en conservant sa surface égale à 2. Héron propose de construire progressivement ce carré en partant de notre rectangle de côtés 1 et 2.

On part donc d'un encadrement de racine 2 par deux nombres :  $a_1 = 1$  et  $b_1 = 2$ , correspondant respectivement à la largeur et longueur du rectangle.

L'objectif est de calculer des valeurs successives de la largeur et de la longueur d'un rectangle dont la forme se rapproche peu à peu du carré recherché. On obtient  $b_2$  en calculant la moyenne de nos deux bornes initiales ( $a_1$  et  $b_1$ ), soit  $b_2 = \frac{3}{2}$ . Puis, on déduit la valeur exacte de  $a_2$  sachant que  $a_2 \times b_2 = 2$ , soit  $a_2 = \frac{4}{3}$ .

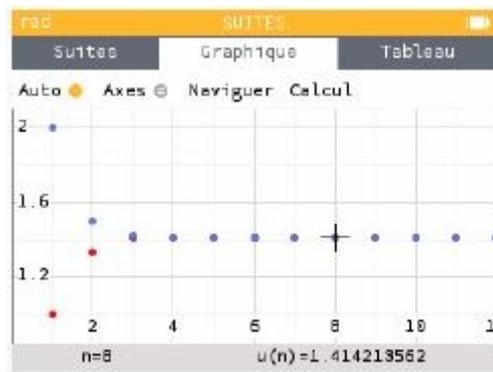
On procède de la même façon et on obtient :  $b_3 = \frac{17}{6}$  et  $a_3 = \frac{12}{17}$ .

A présent, on peut proposer une forme récurrente des suites ( $a_n$ ) et ( $b_n$ ) qui correspondent à notre algorithme.

On admet que les deux suites sont strictement positives pour tout entier naturel  $n > 0$  :

$$\begin{cases} b_1 = 2 \\ b_{n+1} = \frac{a_n + b_n}{2} \end{cases} \text{ et } \begin{cases} a_1 = 1 \\ a_{n+1} = \frac{2}{b_n} = \frac{4}{a_n + b_n} \end{cases}$$

En utilisant la calculatrice pour le graphique des deux suites ( $a_n$ ) et ( $b_n$ ), on remarque que les deux suites semblent converger vers le même nombre et que l'on obtient une approximation de ce nombre dès le rang  $n = 8$ .



### d) Script Python

```
# sans compteur
def racine_de_deux_heron(n) :
    e=10**(-n)
    largeur = 1
    longueur = 2
    while longueur-largeur > e :
        longueur = (longueur+largeur)/2
        largeur = 2/longueur
    return largeur

# avec compteur
def racine_de_deux_heron(n) :
    e=10**(-n)
    c=0
    largeur = 1
    longueur = 2
    while longueur-largeur > e :
        longueur = (longueur+largeur)/2
        largeur = 2/longueur
        c+=1
    return largeur, "le nombre d'itération effectuées est", c

racine_de_deux_heron(5)
(1.41421143847487, "le nombre d'itération effectuées est", 3)
```

**Prolongement possible :**

La méthode de Héron peut permettre également de déterminer la racine cubique de 2. Pour déterminer la racine cubique, nous allons chercher la longueur des côtés d'un cube de volume 2, à partir d'un parallélépipède de même volume mais de hauteur 2, et de base carrée 1.

---

**e) Appréciations et remarques des élèves :**

Après l'étude de la méthode de Héron, les élèves trouvent que cette méthode est assez intuitive et qu'ils auraient pu établir cela eux-mêmes avec un peu plus de réflexion.

Ils ont apprécié pouvoir débattre et élaborer ensemble une méthode qui malgré imprécise s'approche légèrement de l'approximation de Héron.

Ils ont compris qu'un résultat doit être accompagné d'une précision pour pouvoir le mesurer et le comparer à d'autres résultats obtenues par d'autres méthodes.