Exemple d’une activité liée à l’histoire des mathématiques

Extraction d’une racine carrée

Méthode de
Héron d’Alexandrie

Méthode de Héron d'Alexandrie

Point abordés

1. Un point historique
2. Une entrée géométrique

   Niveau 2^{nde}

3. Une entrée algébrique

   Niveau $1^{ère}$

Point historique

Héron d'Alexandrie est un ingénieur, un mécanicien et un mathématicien grec du 1er s. apr. J.-C.

On ne sait pas grand-chose de la vie d'Héron,
si ce n'est qu'il était originaire d’Alexandrie,
au point que les historiens se sont longtemps divisés
sur l'époque à laquelle il avait vécu.

Point historique

Héron d'Alexandrie créa des automates mus par l'eau, s'intéressa à la vapeur et à l'air comprimé.

On lui doit par exemple un projet de machine utilisant la contraction ou la raréfaction de l'air pour ouvrir automatiquement les portes d'un temple.

Point historique

On lui attribue plusieurs formules mathématiques dont :

La formule de Héron :

Elle permet le calcul de l'aire d'un triangle sans utiliser de hauteur

si p = $\frac{a+b+c}{2}$, Aire = $\sqrt{p\ (p-a)\ (p-b)\ (p-c)}$

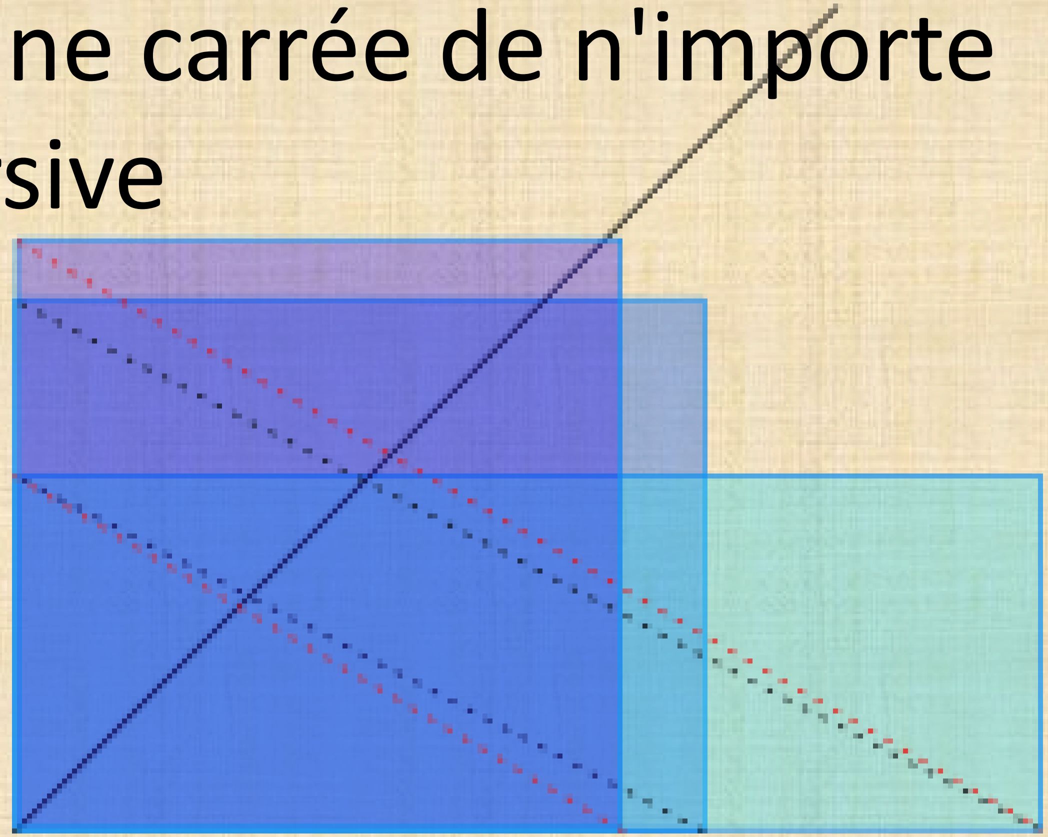
Problème: il faut calculer la racine carrée

Point historique

D’où *<u>la méthode de Héron :</u>*

Elle permet d'approcher la racine carrée de n'importe quel nombre de manière récursive

Texte original



Entrée géométrique

Il est important de rappeler la prégnance de la géométrie dans les mathématiques de l'antiquité.

« Nul n'entre ici, s'il n'est géomètre »

Entrée géométrique

Il est important de rappeler la prégnance de la géométrie dans les mathématiques de l'antiquité.

Il est possible de:

- laisser les élèves décortiquer le texte original,
- ou leur expliquer la méthode,
- ou projeter une vidéo

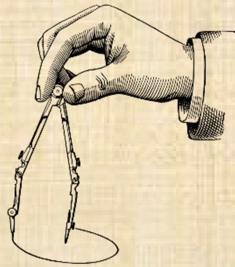
Entrée géométrique

Cherchons à nous approcher de la valeur de $\sqrt{118}$

Partie A :

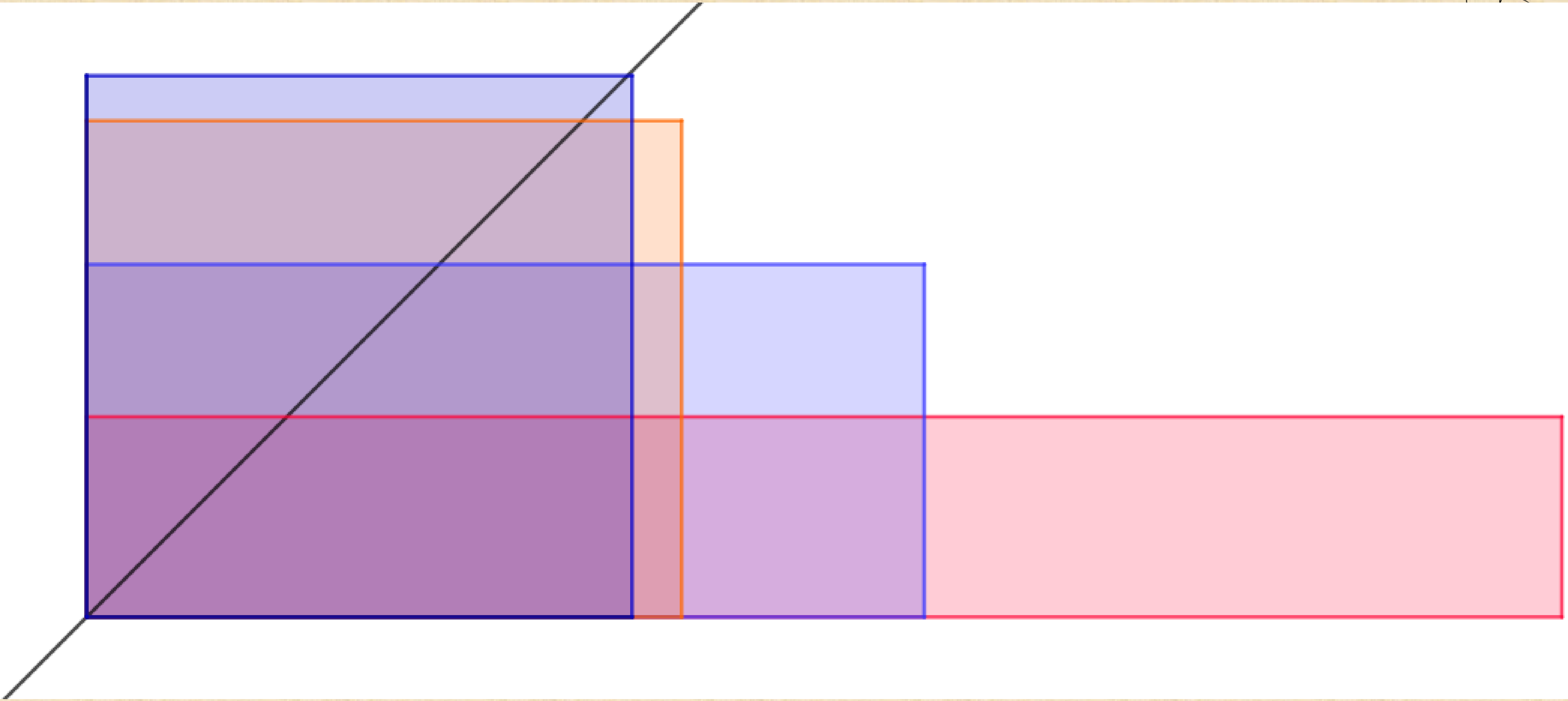
En choisissant une $1^{ère}$ valeur assez éloignée afin que l'évolution des constructions soient plus visibles.

On travaille les algorithmes



Longueur	4	$\frac{67}{4} = 16,75$	$\frac{6377}{536} \approx 11,897$	10,90776569
Largeur	$\frac{118}{4} = 29,5$	$\frac{118}{\frac{67}{4}} = \frac{472}{67}$ $\approx 7,044776$	$\frac{118}{\frac{6377}{536}} \approx 9,91817$	118/10,90776569 ≈ 10,81798082
Moyenne	$\frac{4+\frac{118}{4}}{2} = \frac{67}{4} = 16,75$	$\frac{\frac{67}{4}+\frac{118}{\frac{67}{4}}}{2} = \frac{6377}{536}$ $\approx 11,897$	$\frac{\frac{6377}{536}+\frac{118}{\frac{6377}{536}}}{2}$ $\approx 10,9077$	(L + l) / 2 = 10,86287695
Carré de la moyenne	$\left(\frac{67}{4}\right)^2 = \frac{4489}{16} =$ 280,5625	$\left(\frac{6377}{536}\right)^2 \approx 141,547$	$(10,9077)^2$ $\approx 118,97935$	$10,86287695^2$ $\approx 118,0021$

On peut faire un travail sous géogébra



Entrée géométrique

La véritable méthode de Héron pour le calcul de $\sqrt{118}$

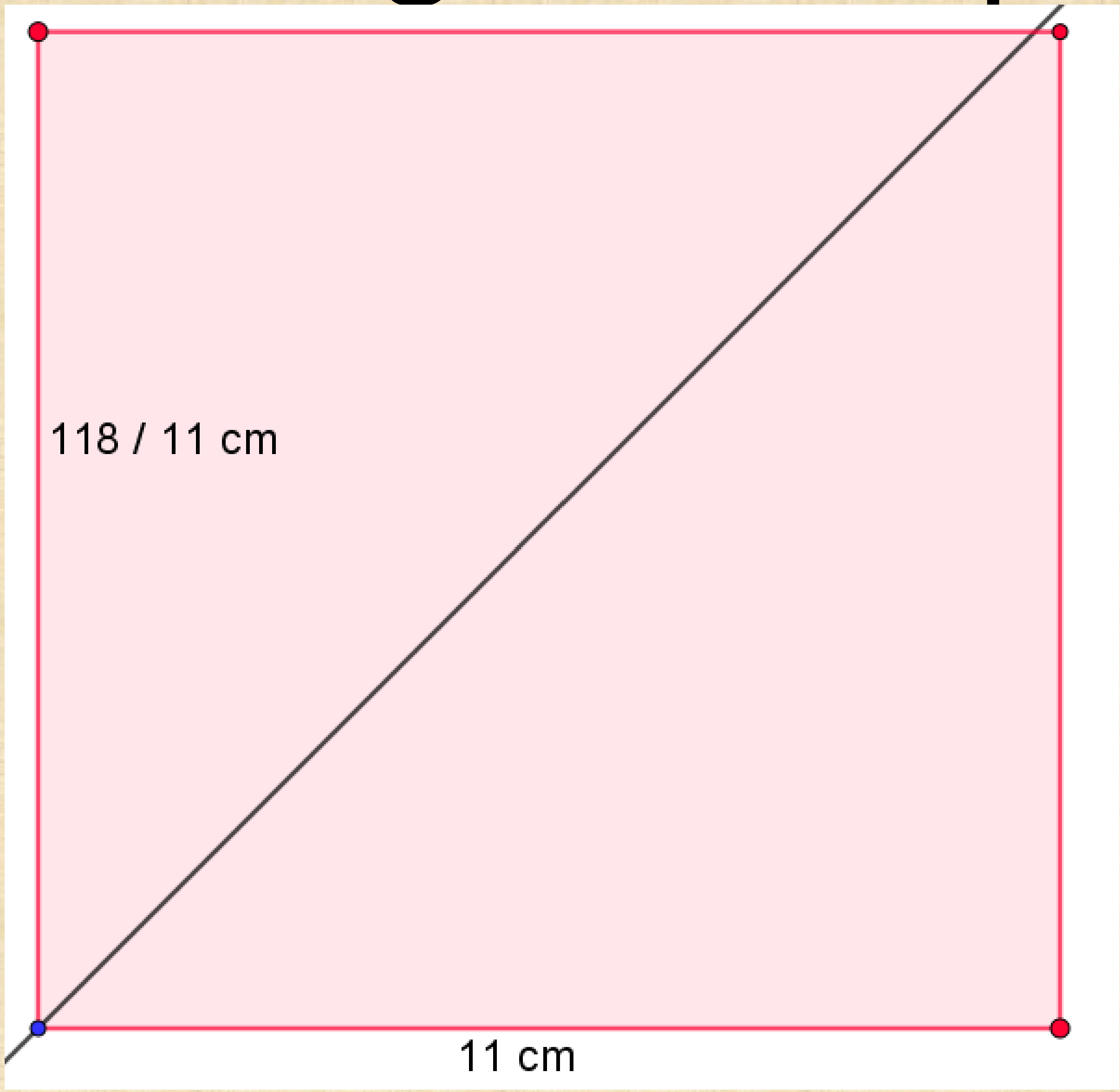
<u>Partie B :</u>

On cherche un carré connu qui soit proche de 118…

121, dont le carré est 11.

On trace donc un rectangle de longueur 11 cm et de largeur 118/11cm

Entrée géométrique



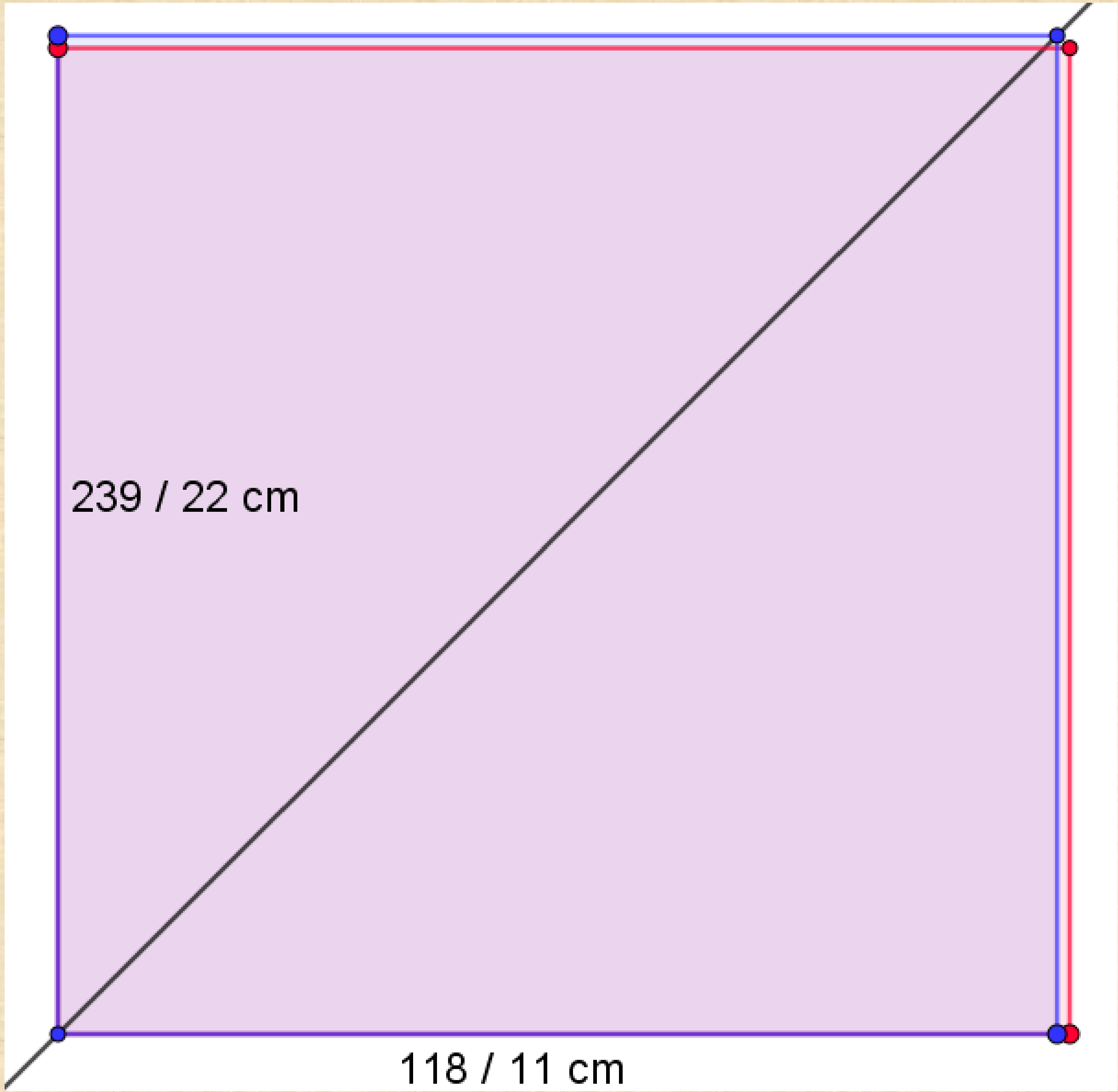
Entrée géométrique

La véritable méthode de Héron pour le calcul de $\sqrt{118}$

Pour tracer une figure « plus carré » on va prendre la valeur moyenne entre 11 et 118/11

$$\frac{11+\frac{118}{11}}{2} = \frac{239}{22} \ (\approx 10{,}863636..)$$

Entrée géométrique



Entrée géométrique

La véritable méthode de Héron pour le calcul de $\sqrt{118}$

En une seule étape, on obtient

$$\left(\frac{239}{22}\right)^2 = \frac{57\,121}{484} = 118{,}018595 \text{ !!!}$$

Entrée géométrique

La convergence est tellement rapide que malgré la construction indépendante de la droite $y = x$,
le sommet du deuxième carrée appartient à cette droite. Le logiciel a atteint sa limite de précision.

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec $a = 720$

Entrée algébrique

Si la méthode géométrique a été faite, on peut directement faire modéliser la méthode de Héron par une suite.

Sinon, on peut traduire le texte afin de calculer le ou les premiers termes.

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Entrée algébrique

Cherchons à nous approcher de la valeur de $\sqrt{118}$

Partie A: *Etude des termes*

Une fois la modélisation faite, plusieurs démarches sont possibles

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n}\right) \end{cases}$$

avec $a = 720$

Entrée algébrique

Cherchons à nous approcher de la valeur de $\sqrt{118}$

- Programmation dans le menu « récurrence » des calculatrices et conjecture
- Algorithme avec les calculatrices
- Algorithme sur Python
(voir fichier Heron d'Alexandrie.py)

A chaque fois, le nombre de décimales proposées est plus important.

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec $a = 720$

Entrée algébrique

```python
# 2nde Extraction d'une racine par la méthode de Héron d'Alexandrie
# Ce programme ne renvoie QUE l'extraction et le nombre d'étapes nécessaires

from math import*

def Heron (a,p):
        # a = valeur du nombre dont on veut la racine
        # p = précision recherchée
    x=a
    f=lambda u:(u+a/u)/2
        # lambda permet de créer directement la fonction f sans
        # la définir au préalable
    y=f(x)
    i=1
    while abs(x-y)>p:
        x=y
        y=f(x)
        i=i+1
    return(y,i) #L'élève doit savoir à quoi correspondent ces deux nombres
```

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec $a = 720$

Entrée algébrique

Console Python

```
*** Console de processus distant Réinitialisée ***
>>>
>>> Heron(118,0.00001)
(10.862780491200215, 8)
>>>
>>>
```

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec $a = 720$

Entrée algébrique

```
# 1ère spé Extraction d'une racine par la méthode de Héron d'Alexandrie
# Ce programme demande à présenter les calculs intermédaires afin de visualiser
#     l'évolution

from math import*

def Heron (a,p):
        # a = valeur du nombre dont on veut la racine
        # p = précision recherchée
    L=[]
    x=a
    f=lambda u:(u+a/u)/2
        # lambda permet de créer directement la fonction f sans la définir
        # au préalable
    y=f(x)
    L=[y]
    while abs(x-y)>p:
        x=y
        y=f(x)
        L.append(y)  # ajoute au fur et à mesure les valeurs de y dans la liste
    return(L)   # Pour avoir le nombre d'étapes, on demande la longueur de la liste
```

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Entrée algébrique

Console Python

```
*** Console de processus distant Réinitialisée ***
>>>
>>> Heron(118,0.00001)
[59.5,
 30.741596638655462,
 17.290021991848747,
 12.057383752177738,
 10.921959040231616,
 10.862940815032808,
 10.862780492383308,
 10.862780491200215]
>>> 
```

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Si on souhaite que la console donne la longueur de la liste…..

```
>>> Heron(118,0.00001)
[59.5,
 30.741596638655462,
 17.290021991848747,
 12.05738375217738,
 10.921959040231616,
 10.862940815032808,
 10.862780492383308,
 10.862780491200215]
>>> L=Heron(118,0.00001)
>>> L
[59.5,
 30.741596638655462,
 17.290021991848747,
 12.05738375217738,
 10.921959040231616,
 10.862940815032808,
 10.862780492383308,
 10.862780491200215]
>>> len(L)
8
```

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Entrée algébrique

$\sqrt{118}$

Partie B: *Etude de la convergence*

- graphiquement,
- algébriquement,

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Etude graphique de la convergence

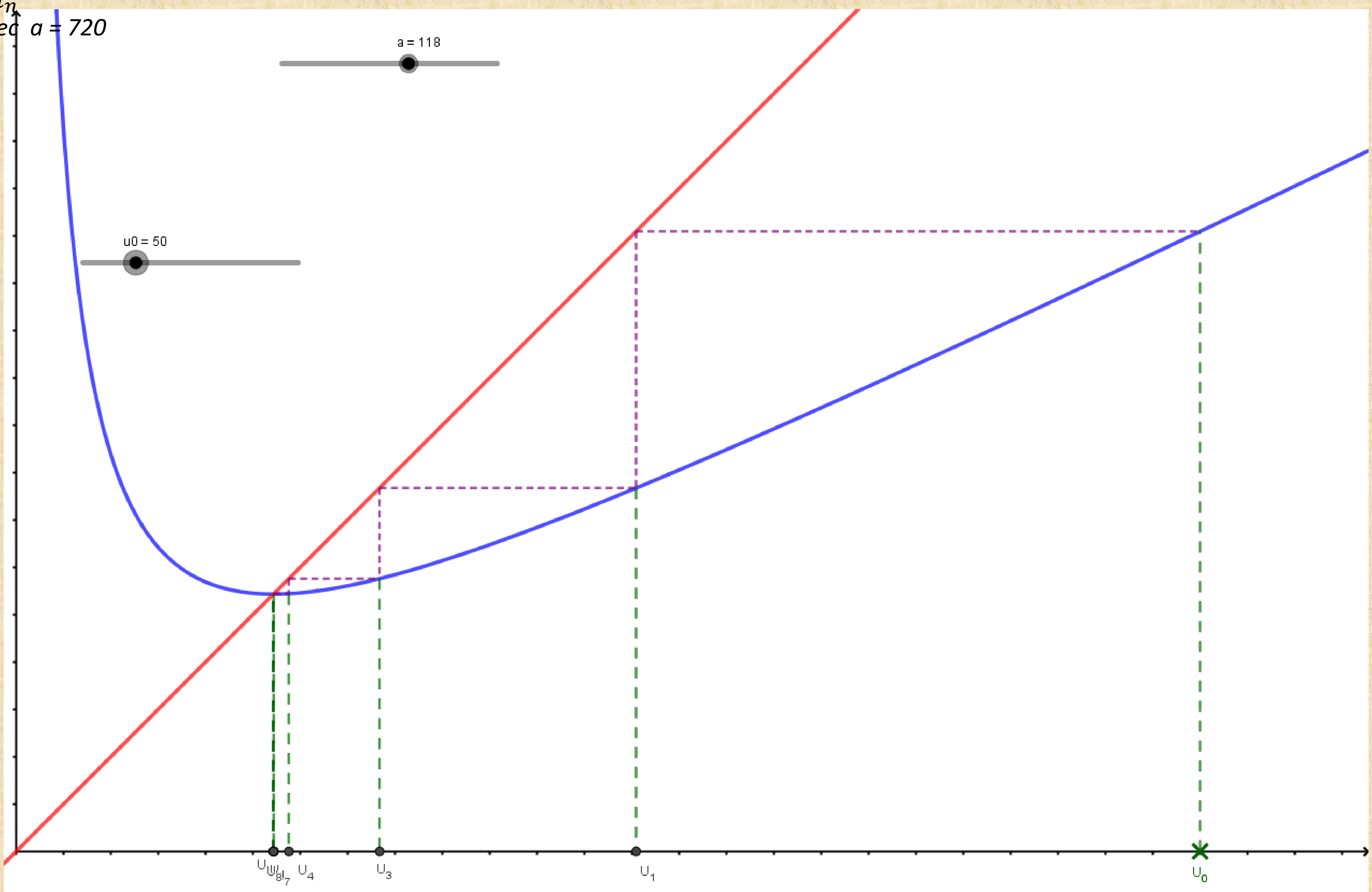
Soit f la fonction définie par : $f(x) = \frac{1}{2}\left(x + \frac{a}{x} \right)$.

La représentation en escalier de la suite donne ainsi avec l'aide du logiciel géogébra :

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Etude graphique de la convergence



[fichier](#) geogébra

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec $a = 720$

Etude algébrique de la convergence

Par exemple avec l'algorithme de Python

Console Python

```
*** Console de processus
>>>
>>> Heron(118,10**(-6))
[59.5,
 30.741596638655462,
 17.29002199184 8747,
 12.057383752177 38,
 10.92195904023161 6,
 10.862940815032808,
 10.862780492383308,
 10.862780491200215]
>>>
```

Console Python

```
*** Console de processus
>>>
>>> Heron(118,10**(-12))
[59.5,
 30.741596638655462,
 17.290021991848747,
 12.05738375217738,
 10.921959040231616,
 10.862940815032808,
 10.862780492383308,
 10.862780491200215,
 10.862780491200215]
>>>
```

on faire découvrir que l'on double le nombre de décimales à chaque étape.

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n}\right) \end{cases}$$

avec $a = 720$

Etude algébrique de la convergence

On **peut** faire démontrer que pour $n \geq 1$, $u_n \geq \sqrt{a}$

Puis montrer que si $a \geq 1$

si $0 \leq u_n - \sqrt{a} \leq 10^{-p}$

alors $0 \leq u_{n+1} - \sqrt{a} \leq 10^{-2p}$

On prouve que l'on double le nombre de décimales à chaque étape.

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Petit bilan

Cette activité peut provoquer un effet « Wouahou » grâce à son efficacité compte tenu de l'époque de Héron d'Alexandrie

Elle offre une multitude de différentiations possibles

Simple calculs de termes

Différents niveaux d'algorithmes

Construction graphique

Démonstration(s)

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec $a = 720$

Petit bilan

Les fiches professeurs (histoire, géométrie, algèbre),
Le texte original, l'algorithme Python et le fichier géogébra sont disponibles sur le site académique

$$\begin{cases} u_0 = 27 \\ u_{n+1} = \frac{1}{2}\left(u_n + \frac{a}{u_n} \right) \end{cases}$$

avec a = 720

Sources

-André Seguin, IREM de la réunion,

- Wikipedia,

- Vidéos de *cordierphychi* et *Top Maths*