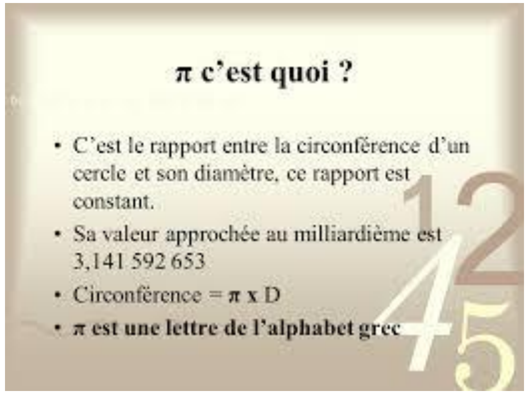
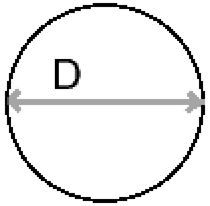
Autour de π

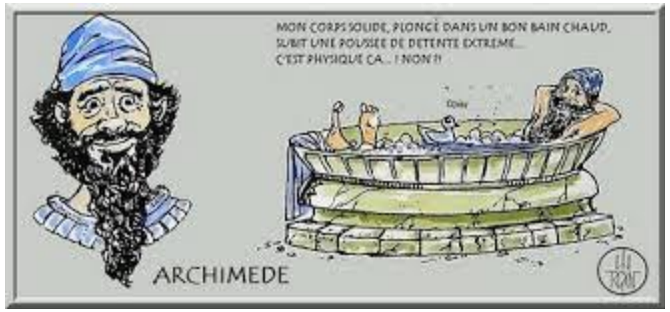


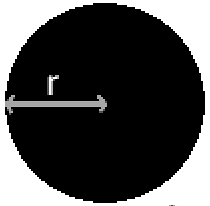
π c'est quoi ?

- C'est le rapport entre la circonférence d'un cercle et son diamètre, ce rapport est constant.
- Sa valeur approchée au milliardième est 3,141 592 653
- Circonférence = **π** x D
- **π est une lettre de l'alphabet grec**

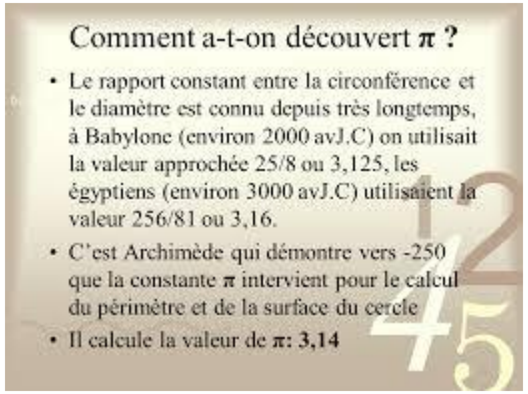


$C = \pi D$





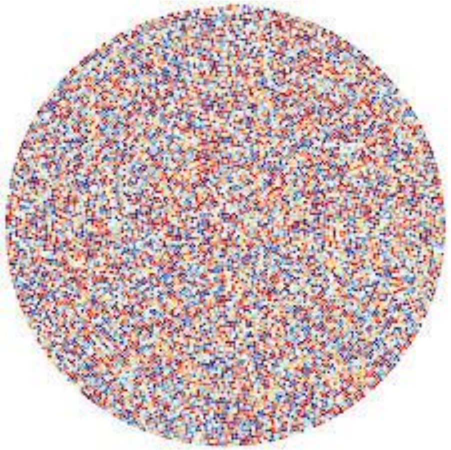
$S = \pi r^2$

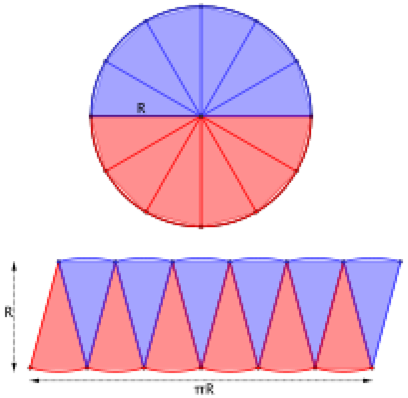


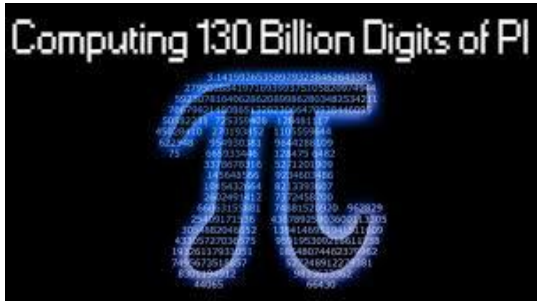
Comment a-t-on découvert **π** ?

- Le rapport constant entre la circonférence et le diamètre est connu depuis très longtemps, à Babylone (environ 2000 avJ.C) on utilisait la valeur approchée 25/8 ou 3,125, les égyptiens (environ 3000 avJ.C) utilisaient la valeur 256/81 ou 3,16.
- C'est Archimède qui démontre vers -250 que la constante **π** intervient pour le calcul du périmètre et de la surface du cercle
- Il calcule la valeur de **π: 3,14**

Leonhard Euler
(1707-1783)







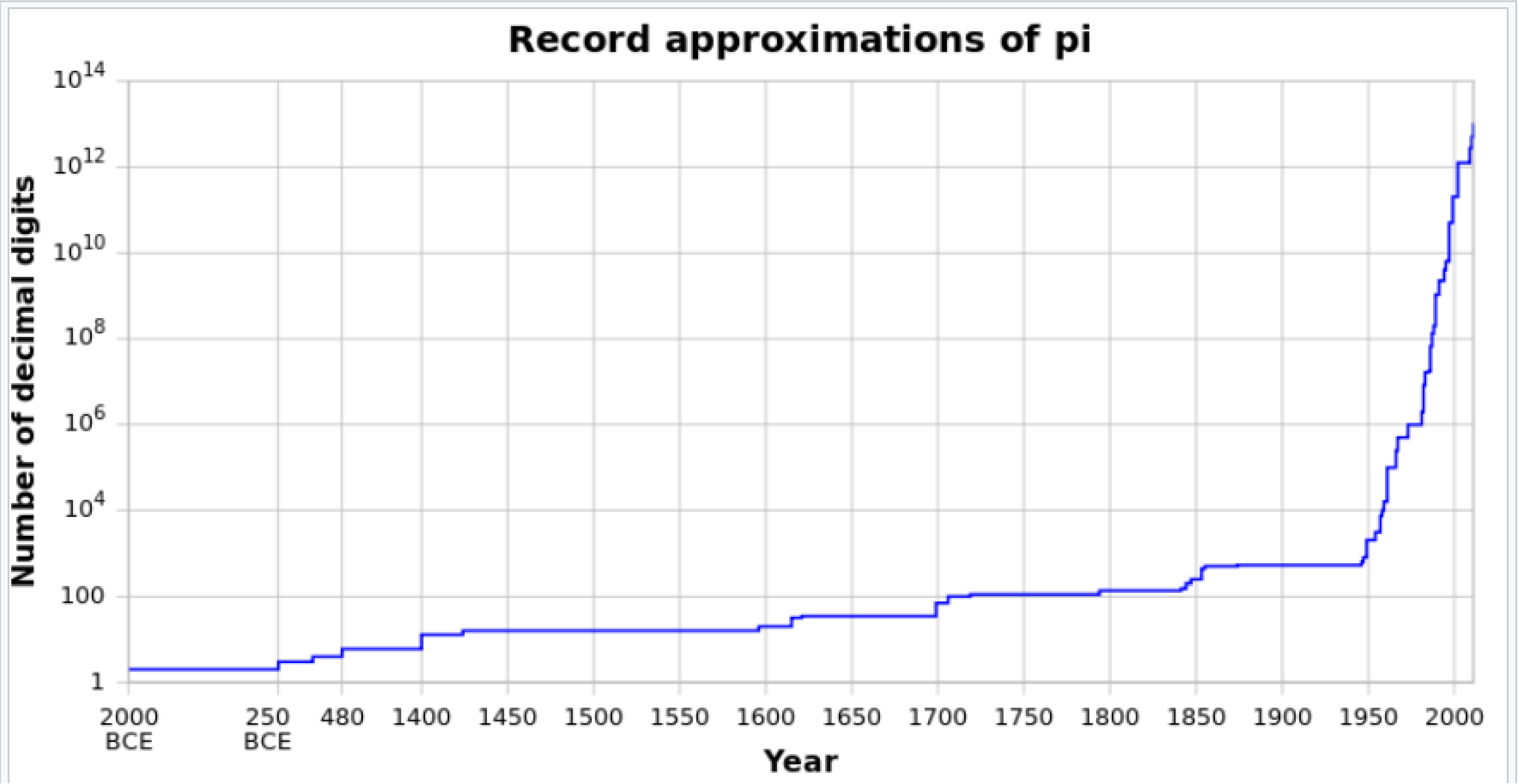
Autour de π

Discovery of Pi

- There is no record how people lived before the discovery of pi and what formulas they used instead of pi.
- Babylonians established the constant circle ratio as 3.125.
- Ahmes, an Egyptian, wrote one of the first known records of Pi on the Rhind Papyrus. He was off by less than 1% of the modern approximation of pi (3.141592).
- Based on Shulba Sutras's observations (India, 600 BC), $\pi \approx 3.088$.

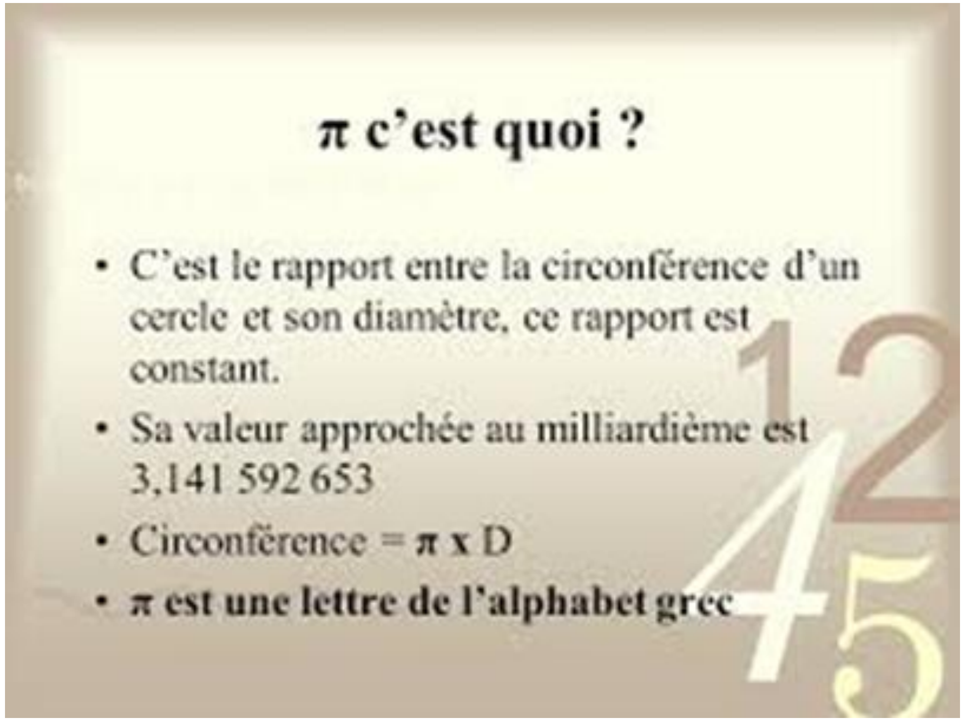
Rhind Papyrus

Autour de π



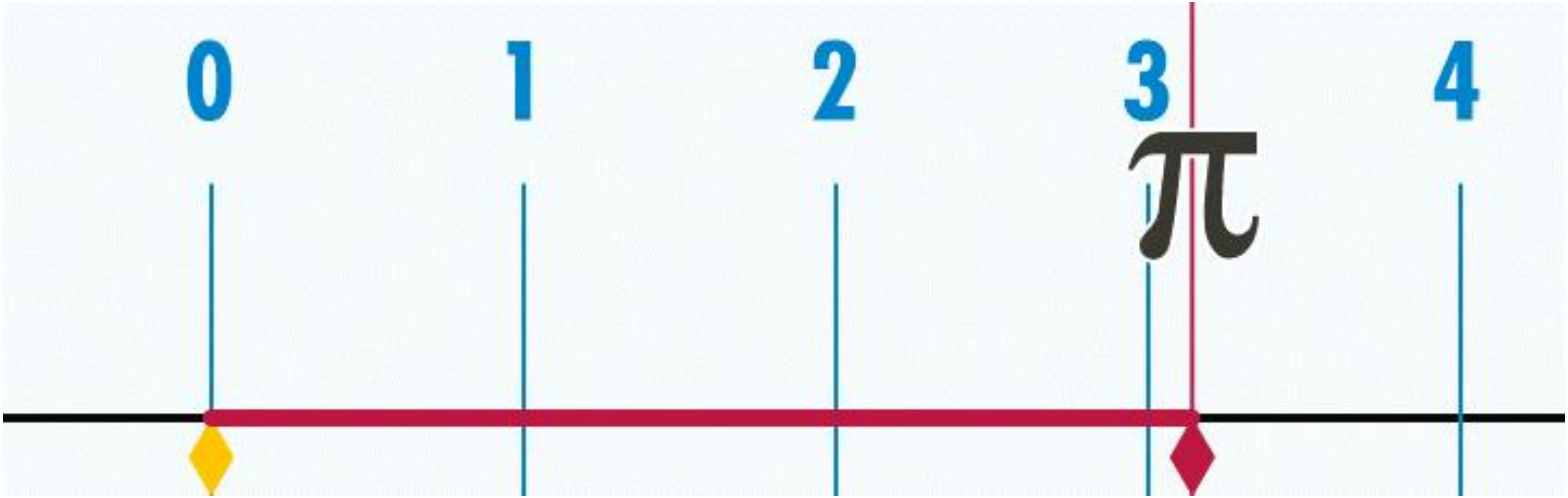
Record approximations of pi
Number of decimal digits
Year
1
100
10^4
10^6
10^8
10^10
10^12
10^14
2000 BCE
250 BCE
480
1400
1450
1500
1550
1600
1650
1700
1750
1800
1850
1900
1950
2000

Autour de π



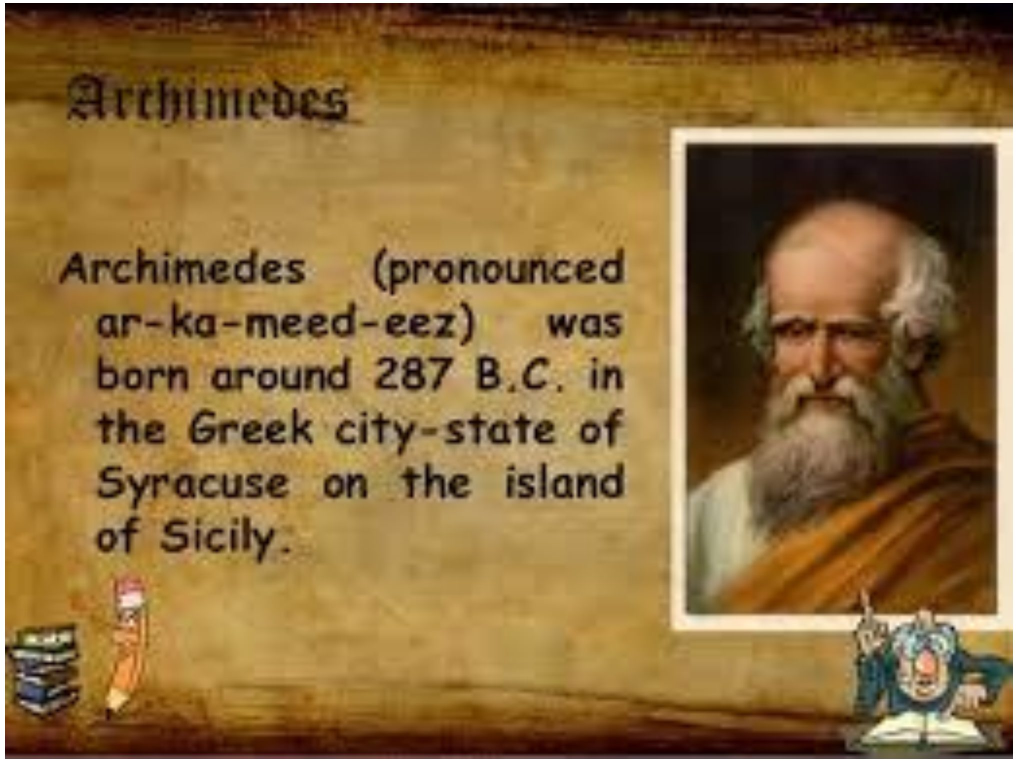
Que j'aime à faire connaître un nombre utile aux sages !
3 1 4 1 5 9 2 6 5 3 5
Immortel Archimède, artiste ingénieur,
8 9 7 9
Qui de ton jugement peut priser la valeur ?
3 2 3 8 4 6 2 6
Pour moi ton problème eut de pareils avantages.
4 3 3 8 3 2 7 9
Tirez circonférence au diamètre etcetera.
5 0 2 8 8
texte cité par BEUTEL 1913

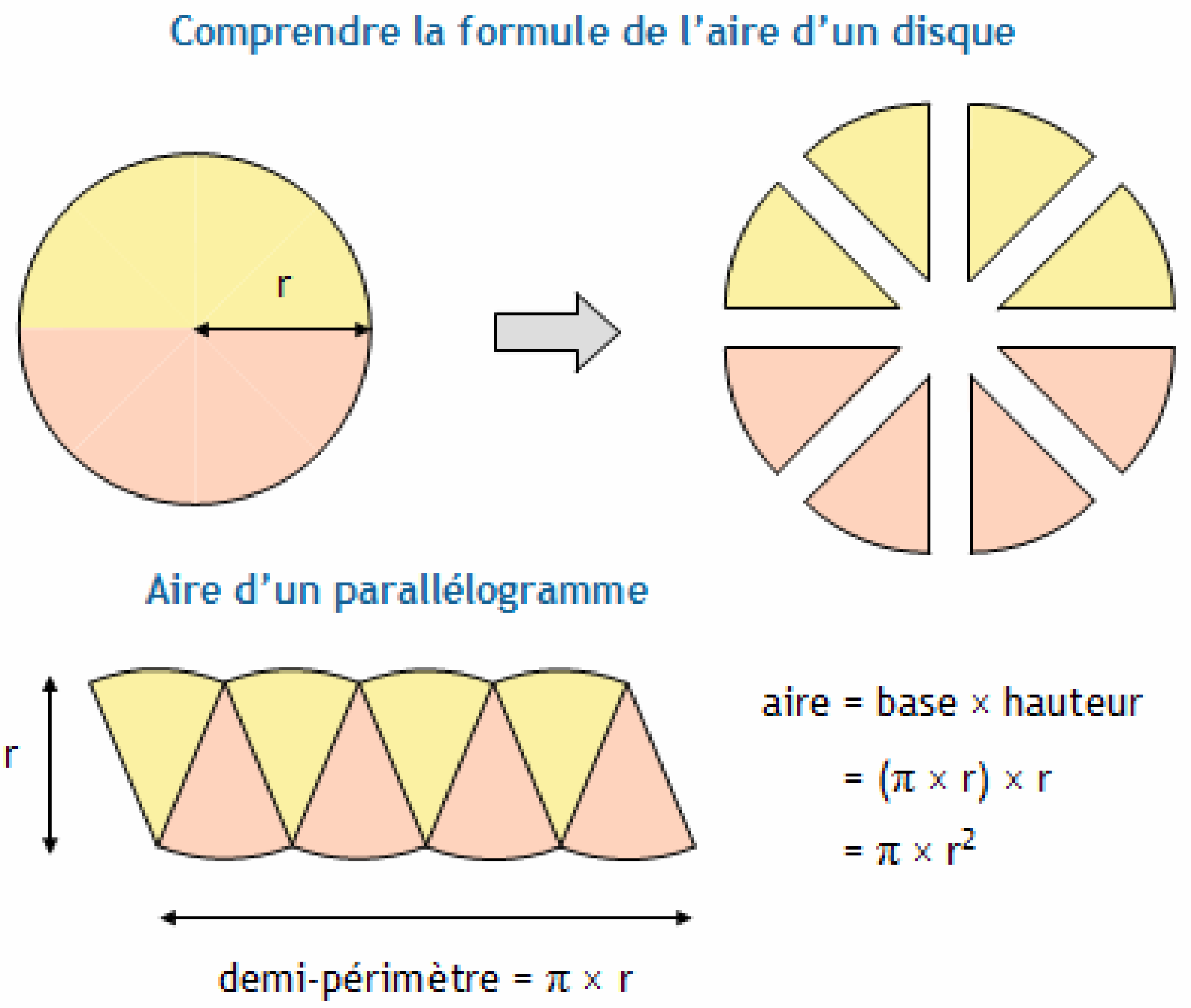
CIRCONFERENCE D'UN CERCLE



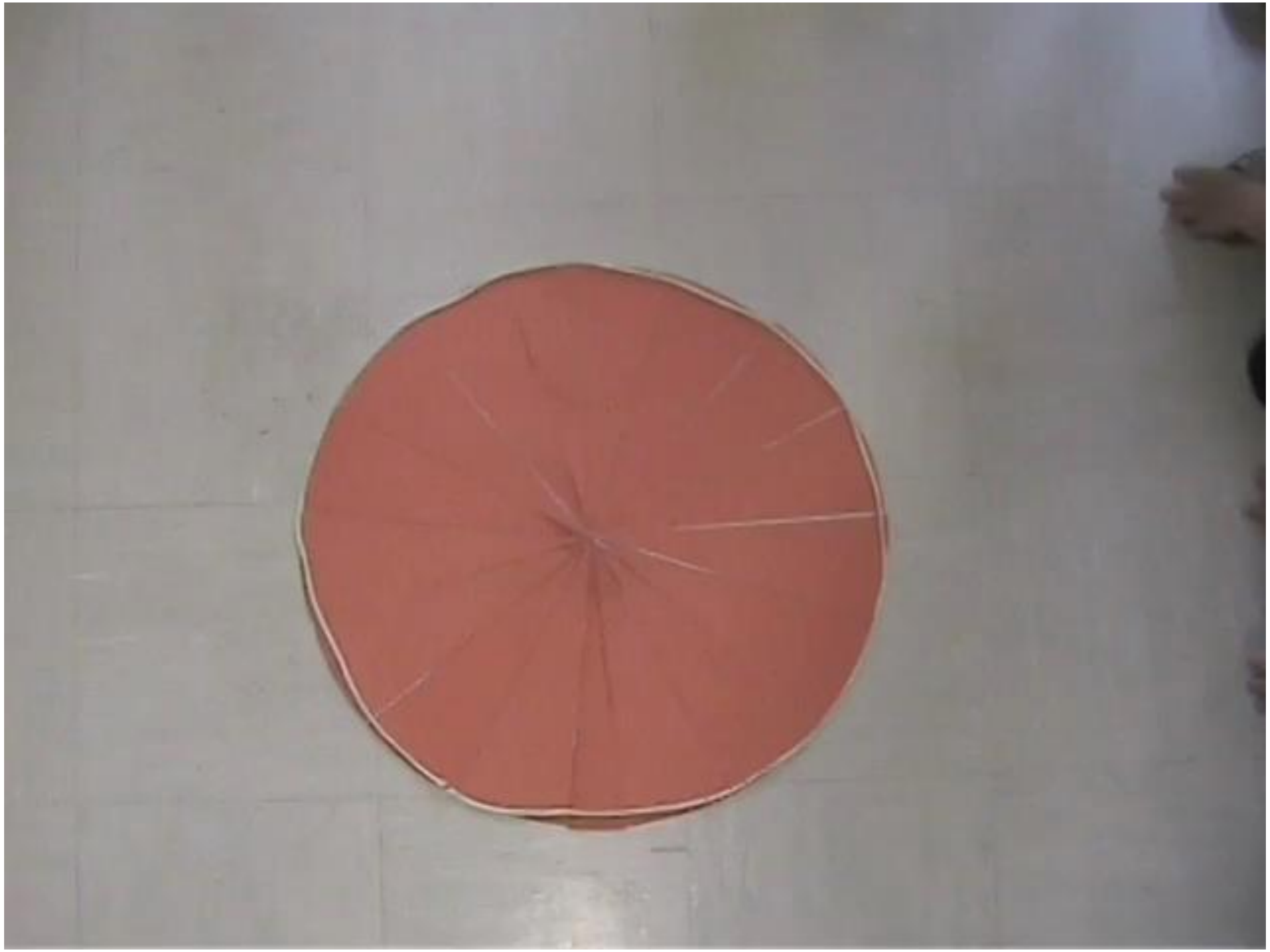
CIRCONFERENCE D'UN CERCLE

AIRE D’UN DISQUE

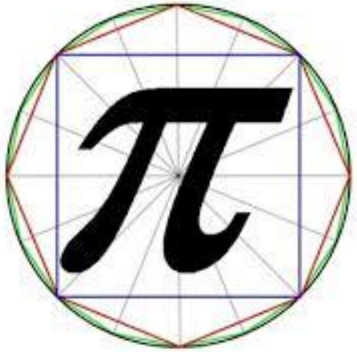


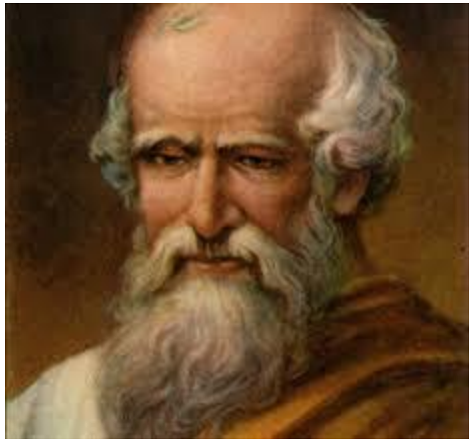


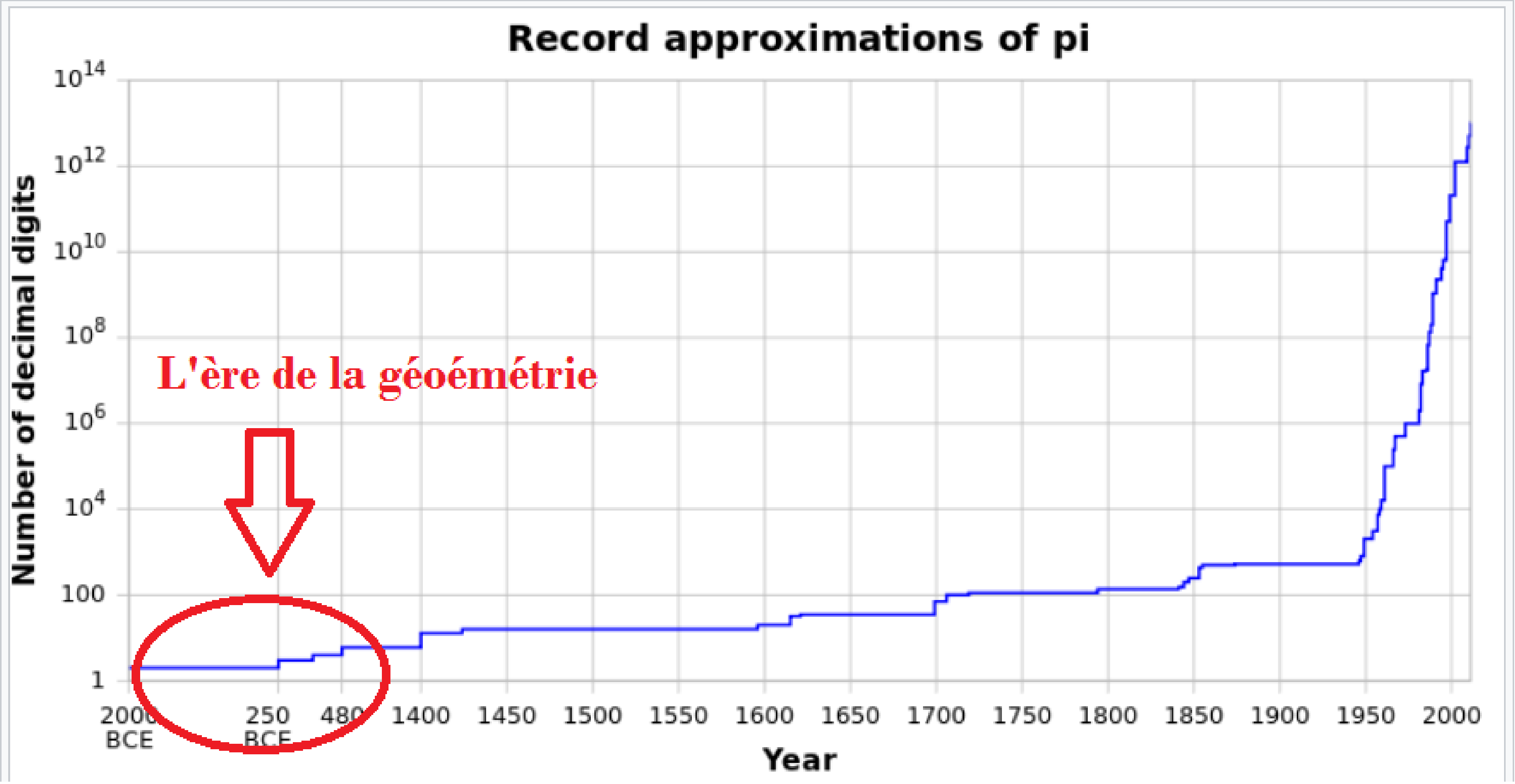
AIRE D'UN DISQUE



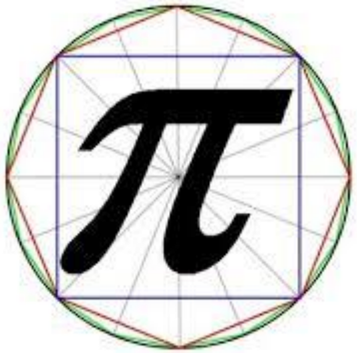
L'ère de la géométrie

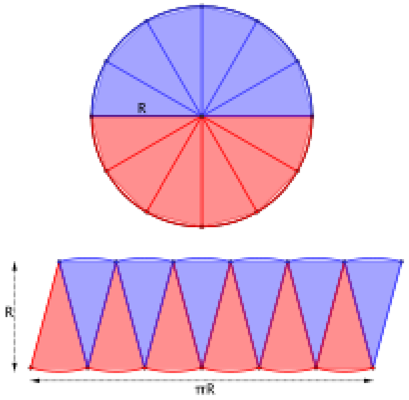




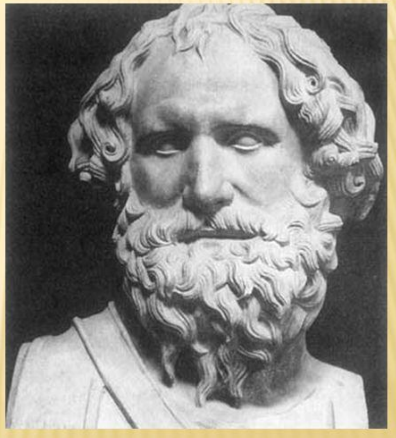


L’ère de la géométrie

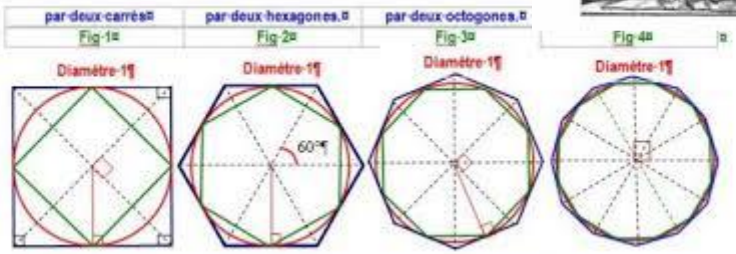


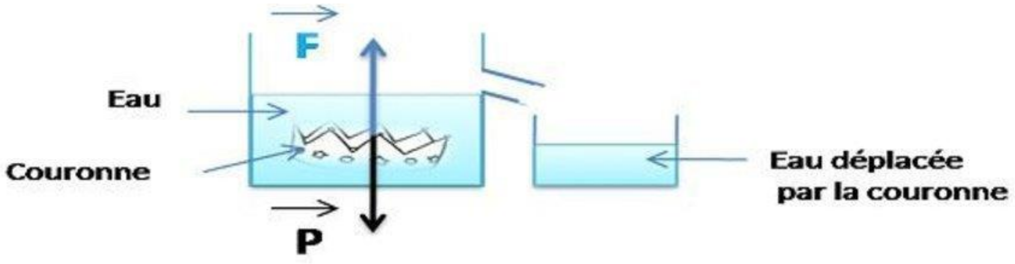


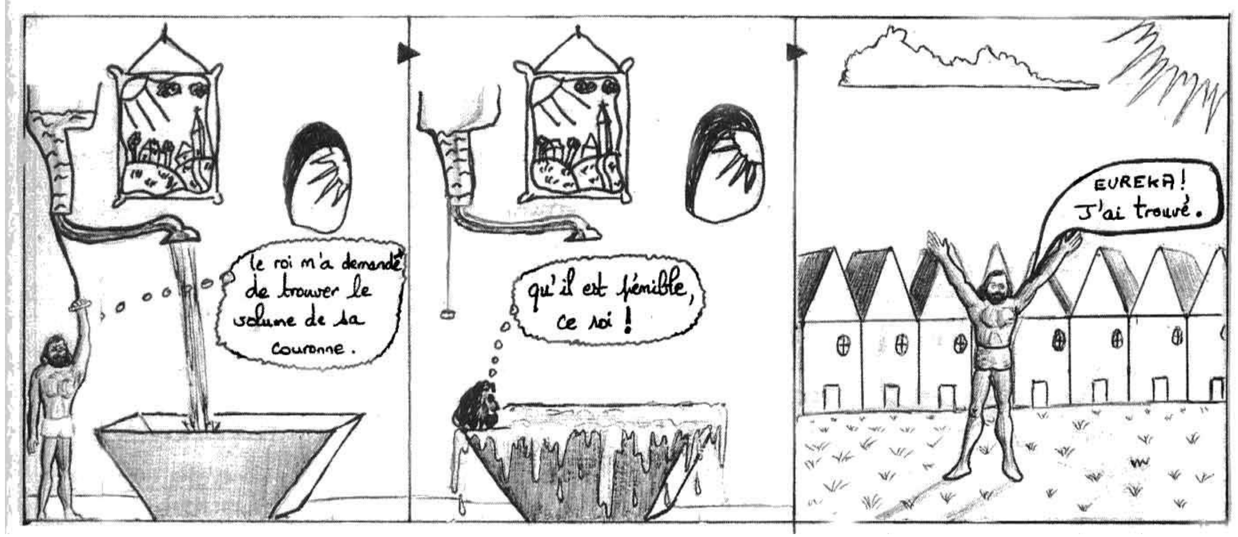
ARCHIMÈDE DE SYRACUSE

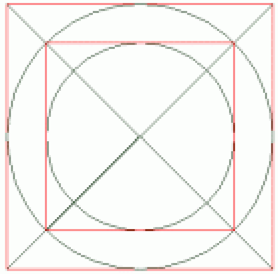


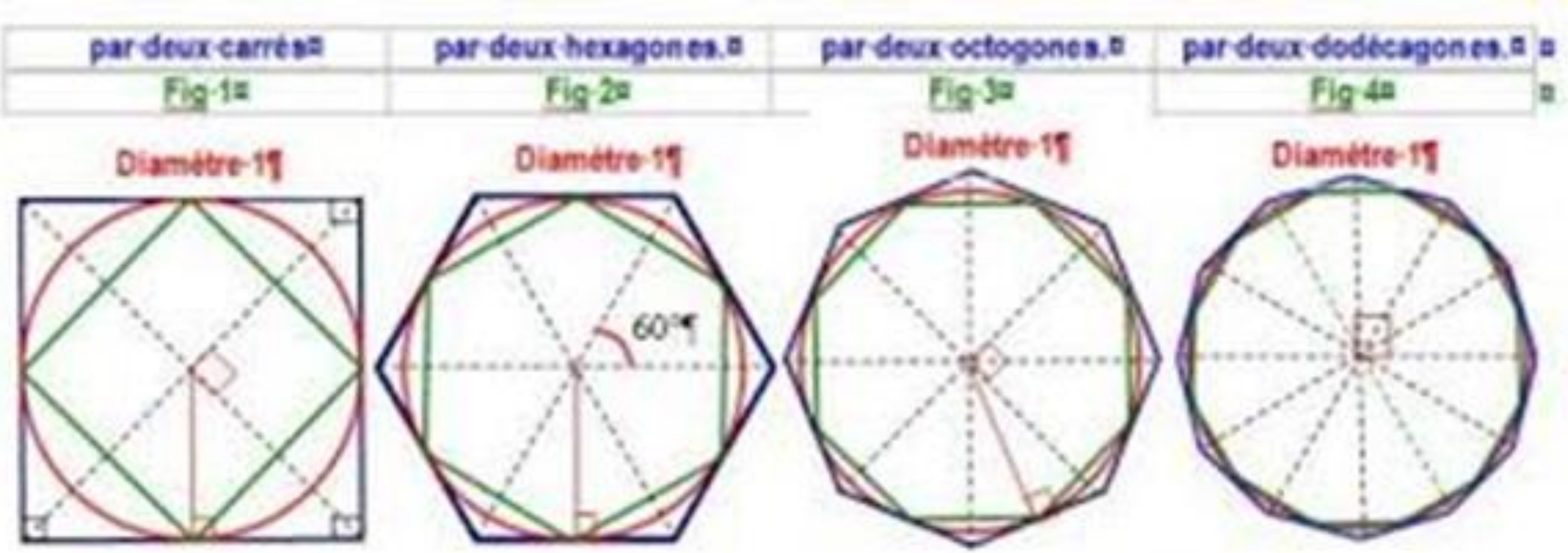
Archimède a été le premier savant à avoir calculer le nombre pi.
Il est né en 287 av J.C. et mort en 212 av J.C. à Syracuse (Italie, Sicile).

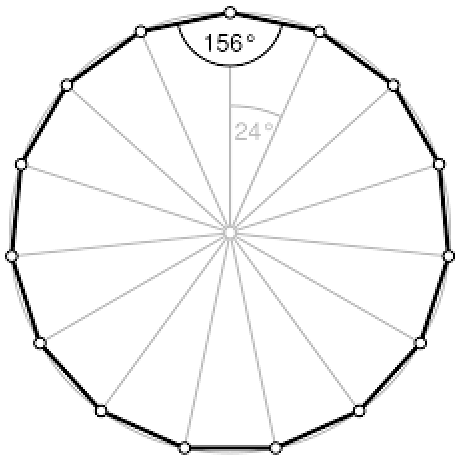


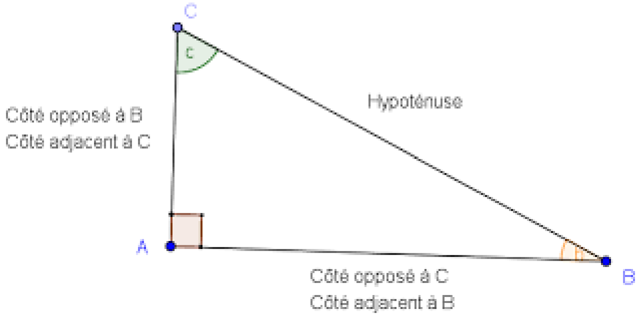








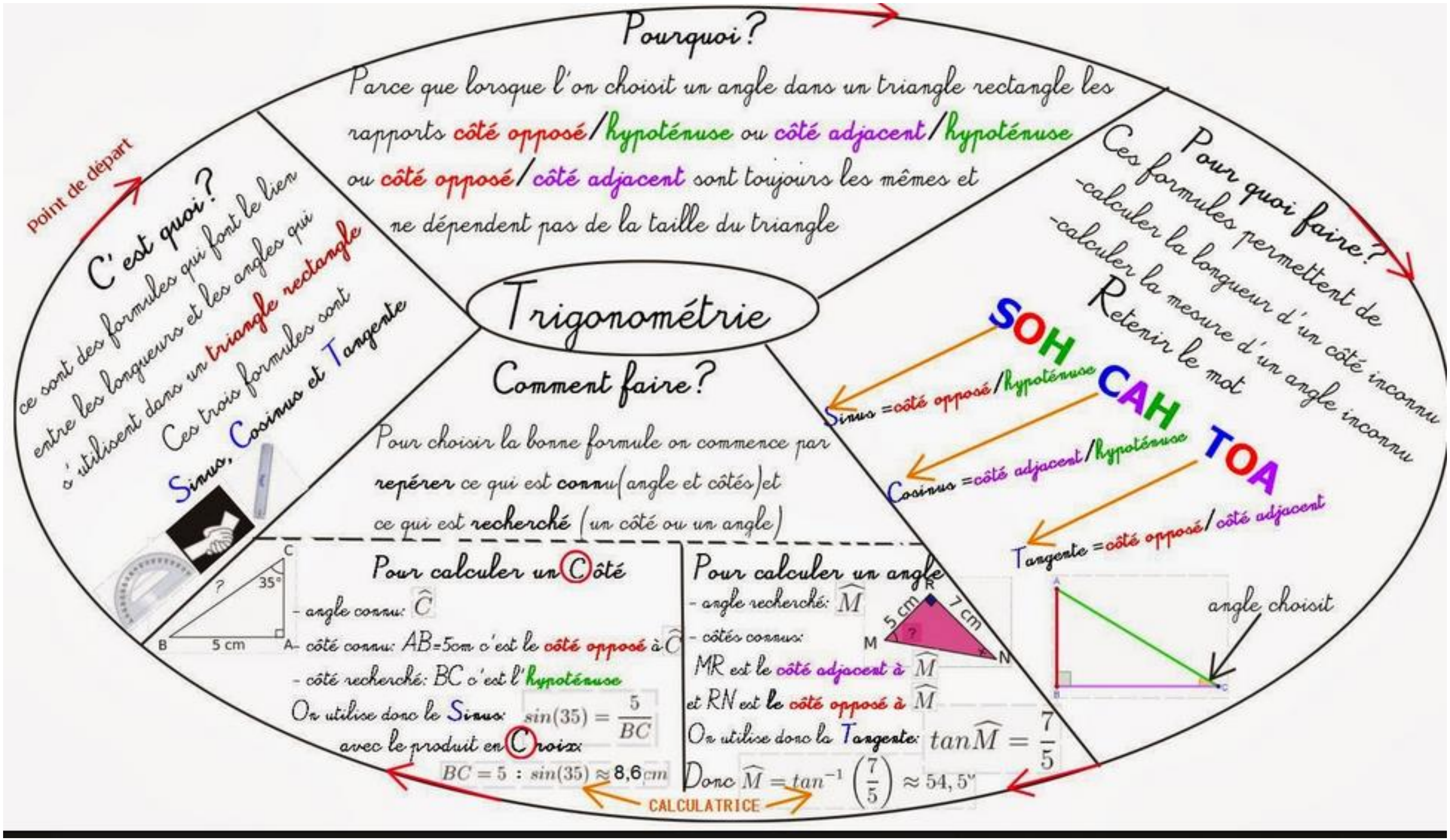




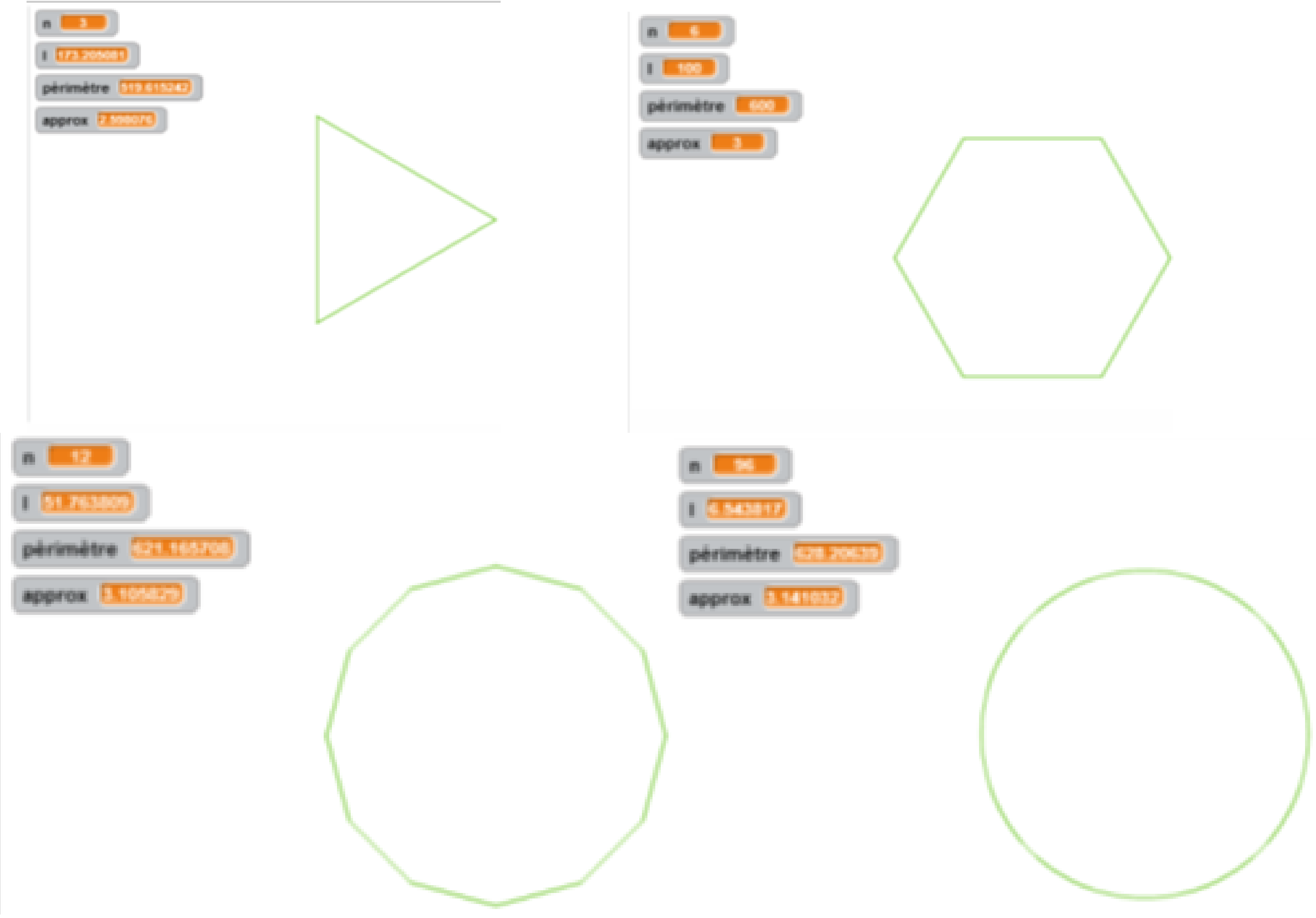
$$\cos(\hat{A}) = \frac{\text{côté adjacent}}{\text{hypoténuse}} = \frac{AB}{AC}$$

$$\sin(\hat{A}) = \frac{\text{côté opposé}}{\text{hypoténuse}} = \frac{BC}{AC}$$

$$\tan(\hat{A}) = \frac{\text{côté opposé}}{\text{côté adjacent}} = \frac{BC}{AB}$$


Trigonométrie
Point de départ
C' est quoi?
ce sont des formules qui font le lien entre les longueurs et les angles qui s'utilisent dans un triangle rectangle
Ces trois formules sont Sinus, Cosinus et Tangente
Pourquoi?
Parce que lorsque l'on choisit un angle dans un triangle rectangle les rapports côté opposé/hypoténuse ou côté adjacent/hypoténuse ou côté opposé/côté adjacent sont toujours les mêmes et ne dépendent pas de la taille du triangle
Pour quoi faire?
Ces formules permettent de
-calculer la longueur d'un côté inconnu
-calculer la mesure d'un angle inconnu
Retenir le mot
SOH CAH TOA
Sinus =côté opposé/hypoténuse
Cosinus =côté adjacent/hypoténuse
Tangente =côté opposé/côté adjacent
angle choisit
Comment faire?
Pour choisir la bonne formule on commence par repérer ce qui est connu (angle et côtés) et ce qui est recherché (un côté ou un angle)
Pour calculer un Côté
- angle connu: $\widehat{C}$
- côté connu: AB=5cm c'est le côté opposé à $\widehat{C}$
- côté recherché: BC c'est l'hypoténuse
On utilise donc le Sinus: $sin(35) = \frac{5}{BC}$
avec le produit en Croix:
$BC = 5 : sin(35) \approx 8{,}6 cm$
C
35°
?
B
5 cm
A
Pour calculer un angle
- angle recherché: $\widehat{M}$
- côtés connus:
MR est le côté adjacent à $\widehat{M}$
et RN est le côté opposé à $\widehat{M}$
On utilise donc la Tangente: $tan\widehat{M} = \frac{7}{5}$
Donc $\widehat{M} = tan^{-1}\left(\frac{7}{5}\right) \approx 54,5°$
R
5 cm
7 cm
M
?
N
CALCULATRICE

Autour de π	La méthode d'Archimède sur Scratch

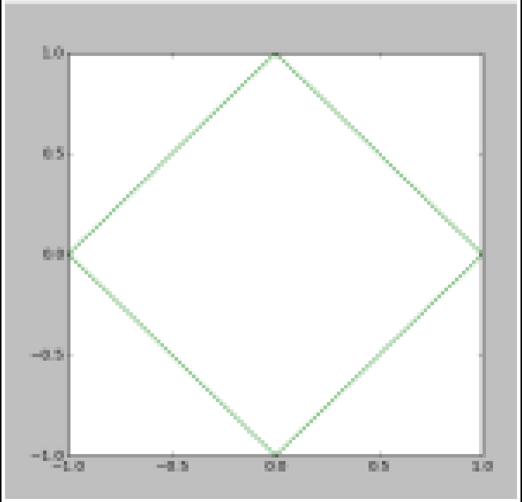


Autour de π | La méthode d'Archimède sur Python

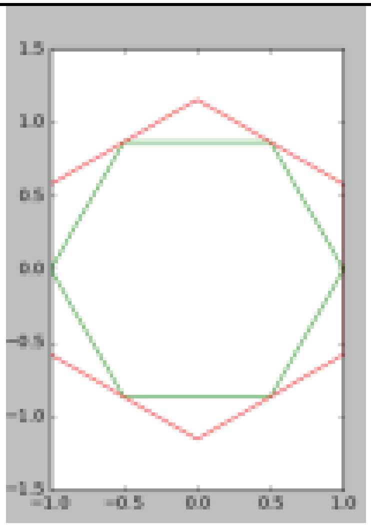
```python
from math import *
import matplotlib.pyplot as plp
def inscrit(n):
    x=1
    y=0
    perimetre=0
    for k in range(n):
        t=x
        u=y
        x=cos(pi*2/n)*x-sin(pi*2/n)*y
        y=cos(pi*2/n)*y+sin(pi*2/n)*t
        perimetre=perimetre+sqrt((x-t)**2+(y-u)**2)
    approx_inf=perimetre/2
    return(approx_inf)
def graph_inscrit(n):
    x=1
    y=0
    X=[x]
    Y=[y]
    for k in range(n):
        t=x
        x=cos(pi*2/n)*x-sin(pi*2/n)*y
        y=cos(pi*2/n)*y+sin(pi*2/n)*t
        X.append(x)
        Y.append(y)
    plp.plot(X,Y,'g-')
    plp.show()
```

```python
def circonscrit(n):
    x=1
    y=tan(pi/n)
    perimetre=0
    for k in range(n):
        t=x
        u=y
        x=cos(pi*2/n)*x-sin(pi*2/n)*y
        y=cos(pi*2/n)*y+sin(pi*2/n)*t
        perimetre=perimetre+sqrt((x-t)**2+(y-u)**2)
    approx_sup=perimetre/2
    return(approx_sup)
def graph_circonscrit(n):
    x=1
    y=tan(pi/n)
    X=[x]
    Y=[y]
    perimetre=0
    for k in range(n):
        t=x
        x=cos(pi*2/n)*x-sin(pi*2/n)*y
        y=cos(pi*2/n)*y+sin(pi*2/n)*t
        X.append(x)
        Y.append(y)
    plp.plot(X,Y,'r-')
    plp.show()
```

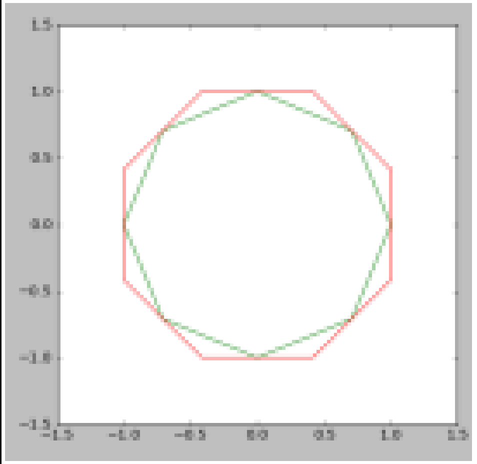
```python
    def graph(n):
        x=1
        y=0
        X_i=[x]
        Y_i=[y]
        for k in range(n):
            t=x
            x=cos(pi*2/n)*x-sin(pi*2/n)*y
            y=cos(pi*2/n)*y+sin(pi*2/n)*t
            X_i.append(x)
            Y_i.append(y)
        x=1
        y=tan(pi/n)
        X=[x]
        Y=[y]
        for k in range(n):
            t=x
            x=cos(pi*2/n)*x-sin(pi*2/n)*y
            y=cos(pi*2/n)*y+sin(pi*2/n)*t
            X.append(x)
            Y.append(y)
        plp.plot(X_i,Y_i,'g-')
        plp.plot(X,Y,'r-')
        plp.show()
    def approx(n):
        approx_inf=inscrit(n)
        approx_sup=circonscrit(n)
        L=[approx_inf,approx_sup]
        return(L)
```



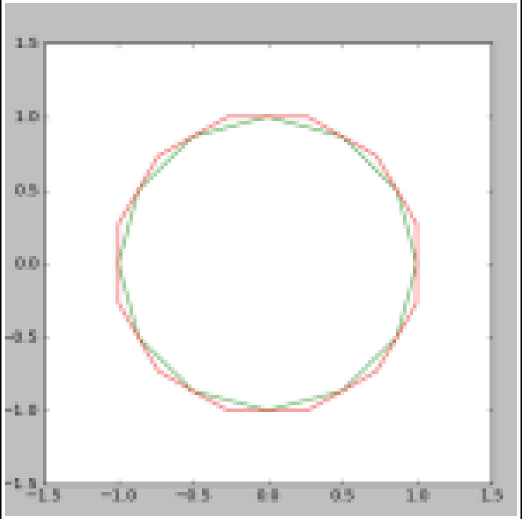
```
>>> approx(4)
[2.82842712474619, 3.9999999999999996]
```



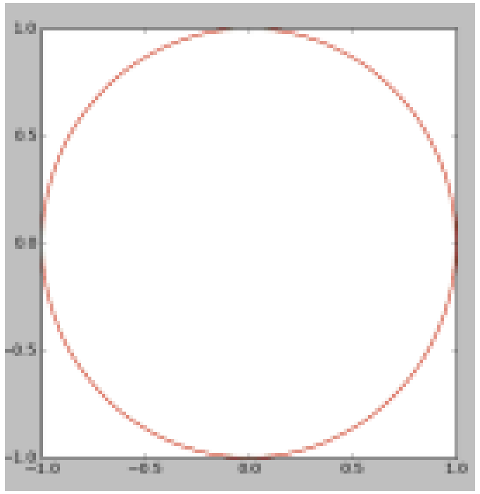
```
>>> approx(6)
[2.9999999999999996, 3.4641016151377535]
```



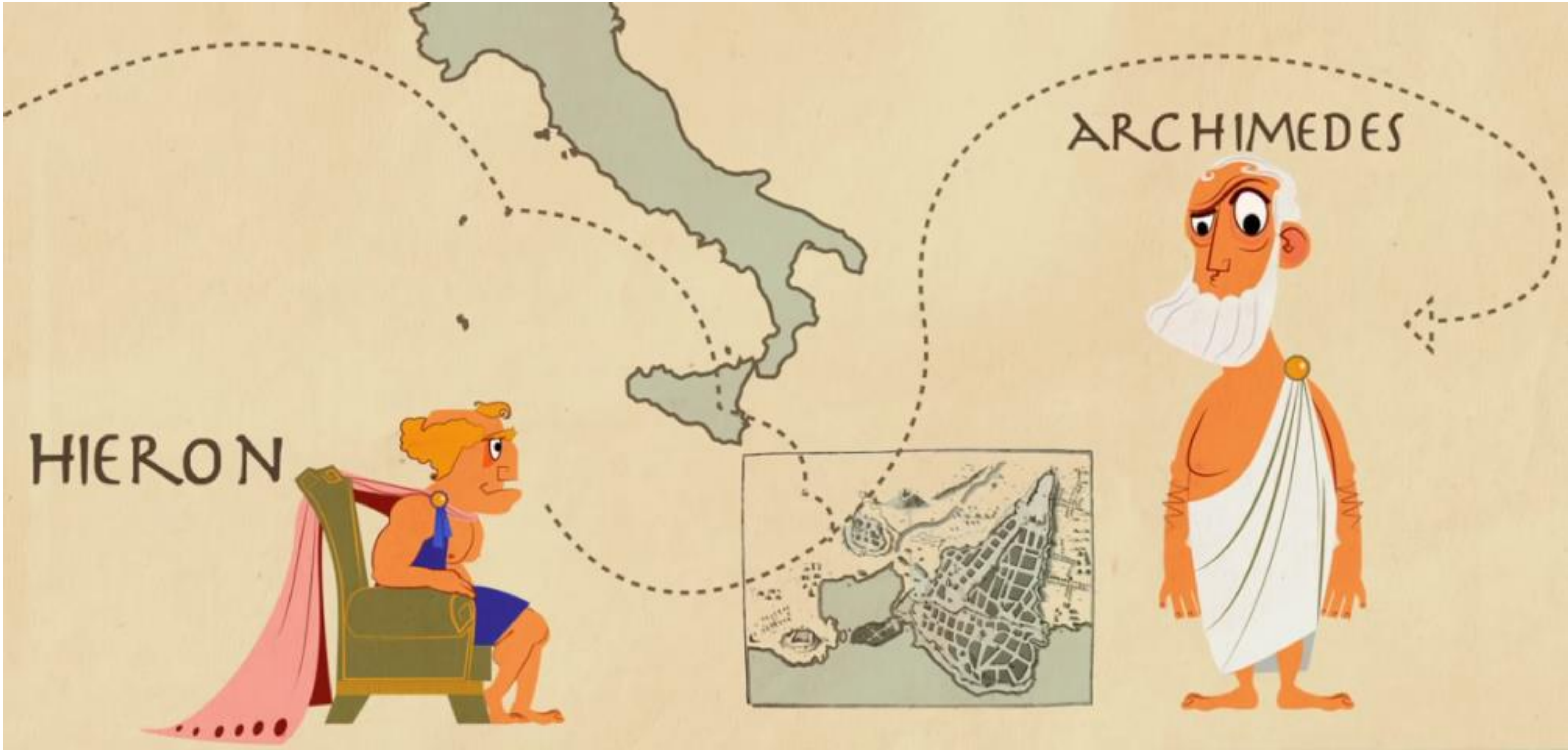
```
>>> approx(8)
[3.0614674589207187, 3.3137084989847676]
```



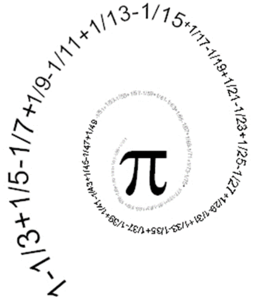
```
>>> approx(12)
[3.105828541230249, 3.215390309173473]
```

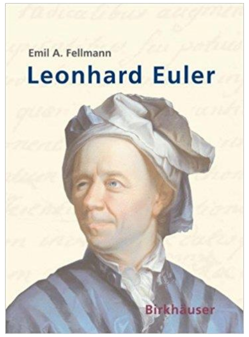


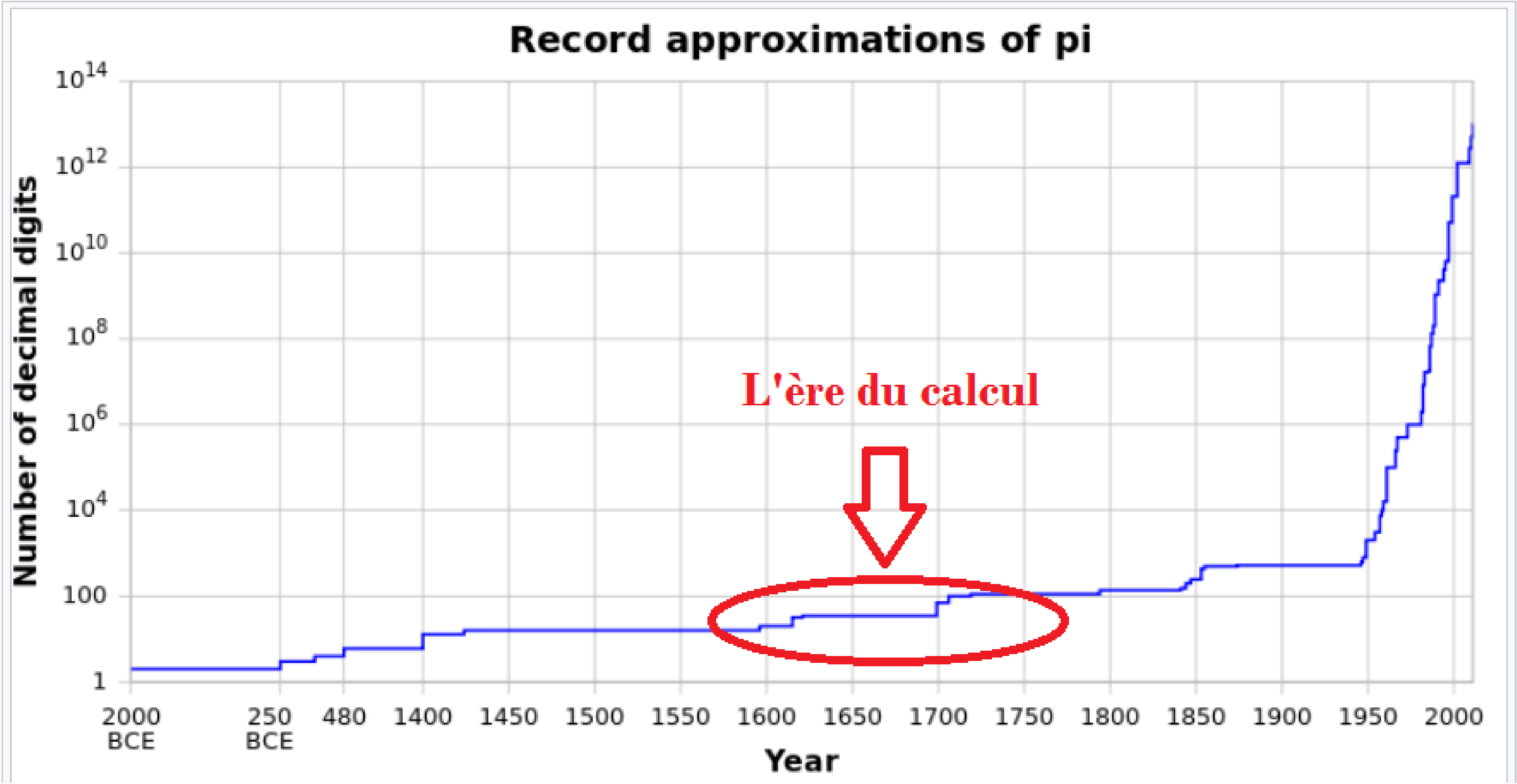
```
>>> approx(96)
[3.141031950890505, 3.1427145996453616]
```


ARCHIMEDES
HIERON

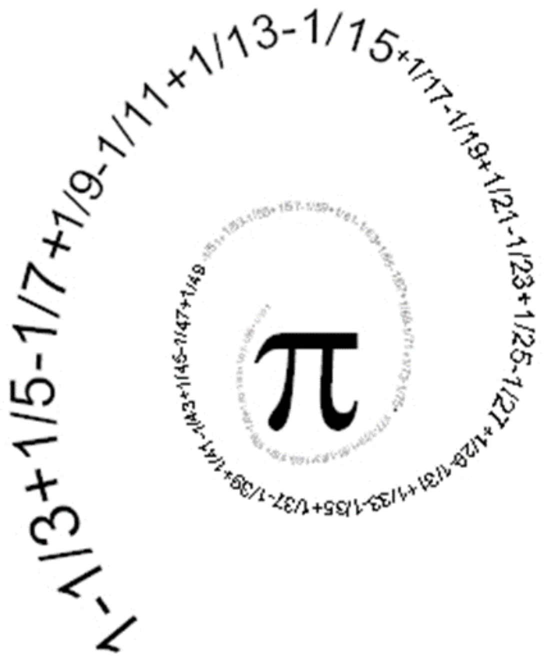
L’ère des calculs







L'ère des calculs

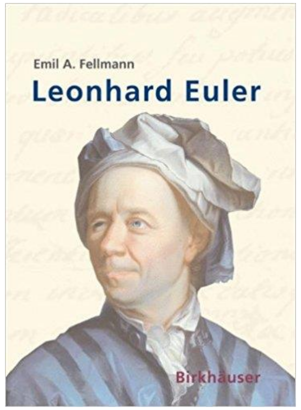


The Wallis Formula For Pi

$$\frac{\pi}{2}=\frac{2}{1}\,\frac{2}{3}\,\frac{4}{3}\,\frac{4}{5}\,\frac{6}{5}\,\frac{6}{7}\cdots$$

Gottfried Wilhelm Leibniz
(1646 - 1716)

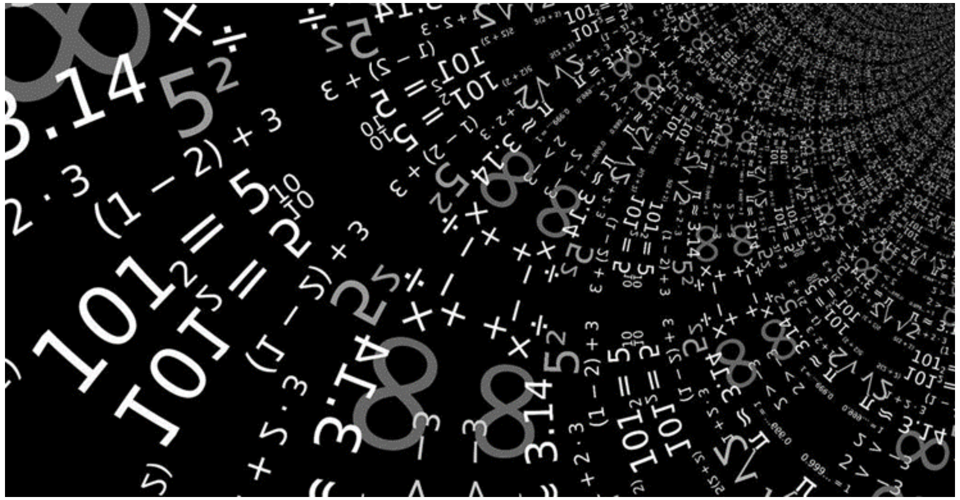
$$\pi = 4\sum_{n=0}^{\infty}\frac{(-1)^n}{(2n+1)}$$



$$\sum_{n=1}^{\infty}\frac{1}{n^2}=\frac{\pi^2}{6}$$

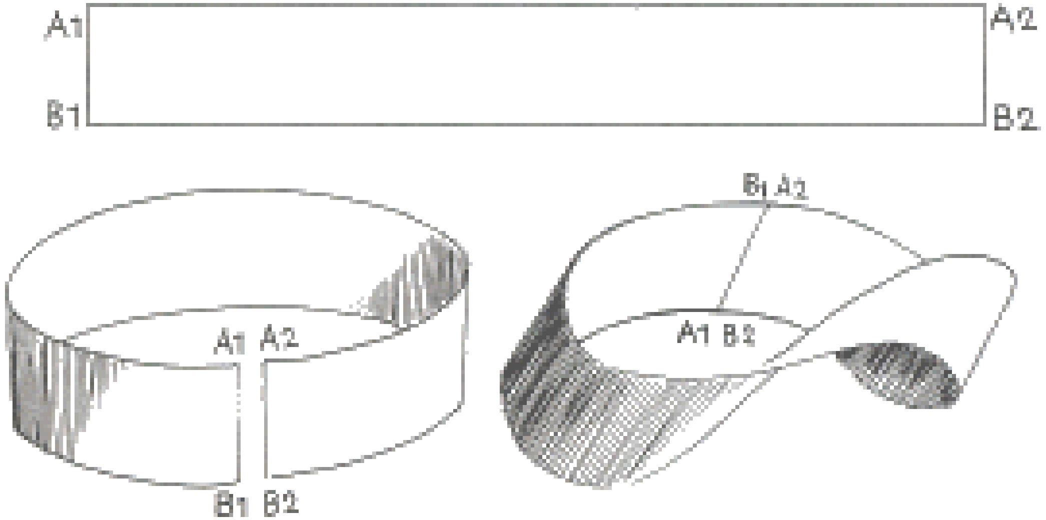
$$\Sigma$$

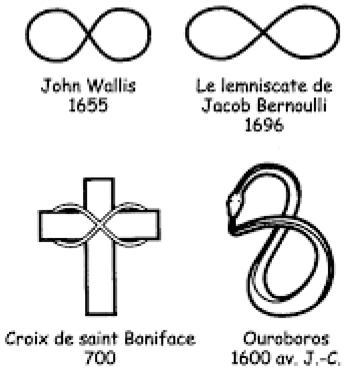
Notation	Notation	Lire
Complète	$s_n=\sum_{i=1}^{n} i^2$	Sigma de i au carré pour i de 1 à n
Abrégée	$\sum n^2$	Sigma n carré

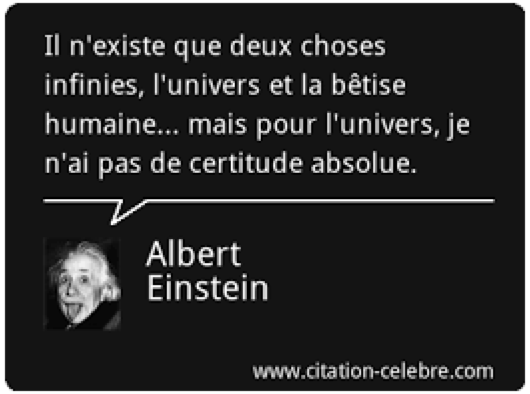


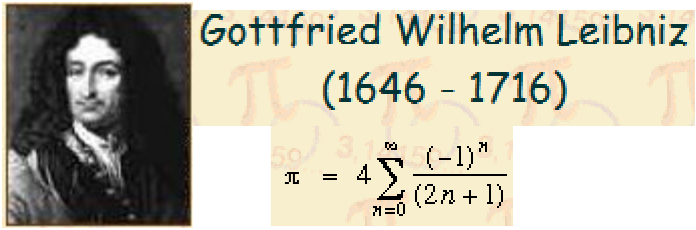
The Wallis Formula For Pi

$$\frac{\pi}{2} = \frac{2}{1}\,\frac{2}{3}\,\frac{4}{3}\,\frac{4}{5}\,\frac{6}{5}\,\frac{6}{7}\cdots$$









À droite, Isaac Newton ; à gauche Gottfried Wilhelm von Leibniz. Les deux plus grands intellectuels de leur temps. Ils sont philosophes, mathématiciens et physiciens. Mais Newton, président de la Royal Society de Londres, est aussi alchimiste, astronome et économiste, tandis que Leibniz, président de l'Académie des sciences de Berlin, est juriste, linguiste, historien, géographe, diplomate et théologien !

Dieu est un océan, dont nous n'avons reçu que quelques gouttes...

Leibniz

www.citation-celebre.com

HISTORIQUE

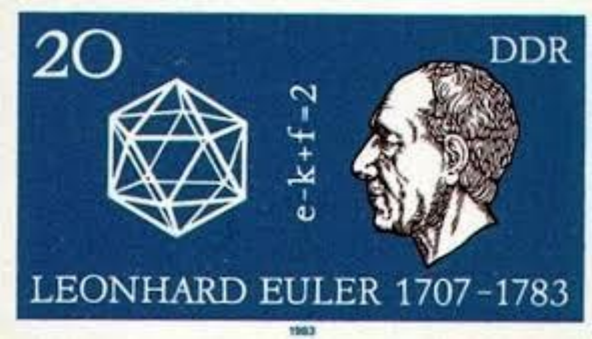
- La création du calcul infinitésimal est liée à une polémique entre deux mathématiciens : Isaac Newton et Gottfried Wilhelm von Leibniz. Néanmoins, son histoire est très vaste, d'Archimède à Barrow en passant par Fermat.

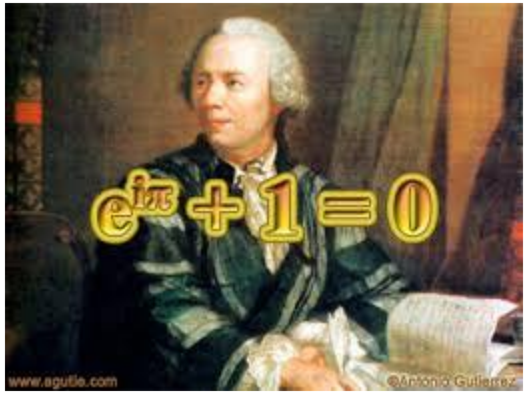
Barrow, Descartes, Fermat, Huygens et Wallis contribuèrent également dans une moindre mesure au développement du calcul infinitésimal.

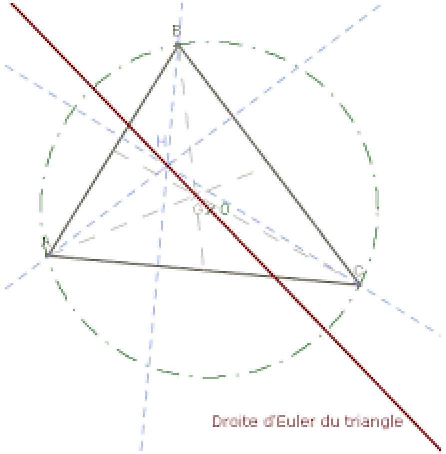
Sir Isaac Newton Gottfried Wilhelm von Leibniz

Leibniz consacre également de nombreuses années à concevoir une machine à calculer capable d'effectuer des multiplications et des divisions.

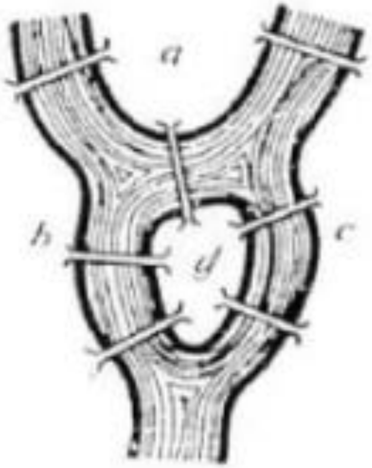
$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

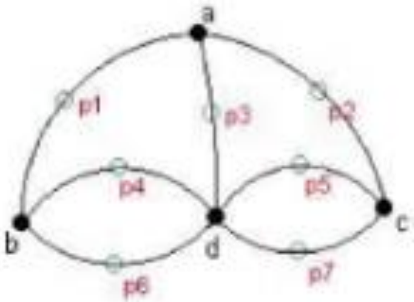






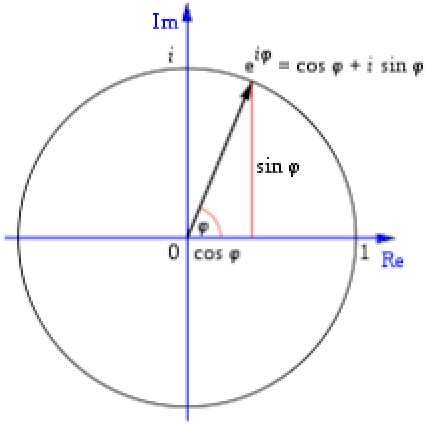
La ville de Königsberg comprenait sept ponts, disposés selon le schéma ci-dessous. Le souhait des habitants de Königsberg était de faire un trajet passant une fois et une seule par chaque pont. Comment faire ?





The Euler characteristic for convex polyhedra always equals 2

Name	Image	Vertices V	Edges E	Faces F	Euler characteristic: $V - E + F$
Tetrahedron		4	6	4	2
Hexahedron or cube		8	12	6	2
Octahedron		6	12	8	2
Dodecahedron		20	30	12	2
Icosahedron		12	30	20	2



```python
from math import *
def Leibniz(n):
    S=0
    for k in range(n+1):
        S=S+(-1)**(k)/(2*k+1)
    approx=4*S
    return(approx)
```

Python 3.5.2 Shell

File Edit Shell Debug Options Window

```
Python 3.5.2 (v3.5.2:4def2a2901a5
D64)] on win32
Type "copyright", "credits" or "l
>>>
 RESTART: C:/Data/1718/dnl premiè
ulation of pi/Leibniz1.py
>>> Leibniz(10)
3.232315809405594
>>> Leibniz(100)
3.1514934010709914
>>> Leibniz(1000)
3.1425916543395442
>>> Leibniz(10000)
3.1416926435905346
>>> Leibniz(100000)
3.1416026534897203
>>>
```

```python
def Wallis(n):
    k=n//2
    Part1=1
    Part2=1
    for i in range(1,k+1,1):
        Part1=Part1*2*i/(2*i-1)
        Part2=Part2*2*i/(2*i+1)
    P=Part1*Part2
    approx=2*P
    return(approx)
```

Python 3.5.2 Shell

File Edit Shell Debug Options Window Help

```
Python 3.5.2 (v3.5.2:4def2a2901a5, Ju
D64)] on win32
Type "copyright", "credits" or "licen
>>>
 RESTART: C:\Data\1718\dnl première\d
ulation of pi\Leibniz1.py
>>> Wallis(10)
3.002175954556908
>>> Wallis(100)
3.126078900215413
>>> Wallis(1000)
3.140023818600602
>>> Wallis(10000)
3.1414355935899456
>>> Wallis(100000)
3.141576945822886
>>>
```

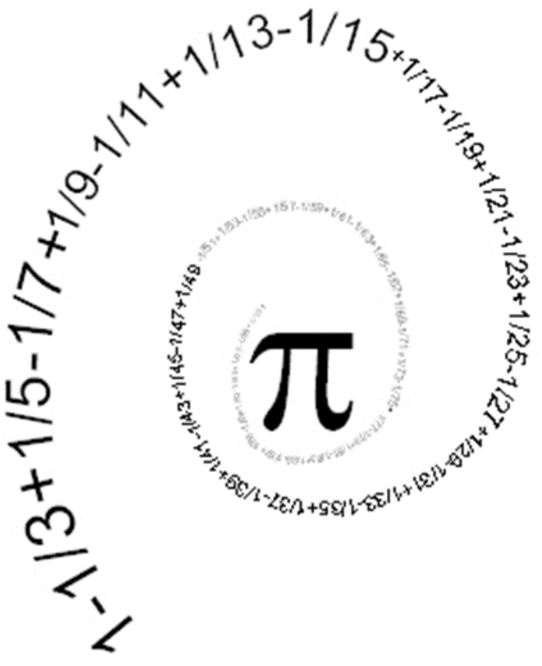
```python
def Euler(n):
    S=0
    for k in range(1,n+1,1):
        S=S+1/(k**2)
    approx=sqrt(6*S)
    return(approx)
```

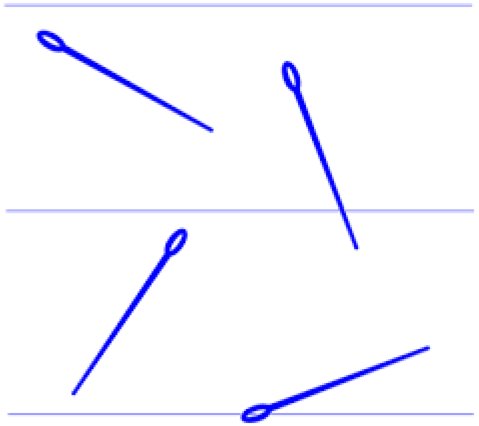
Python 3.5.2 Shell

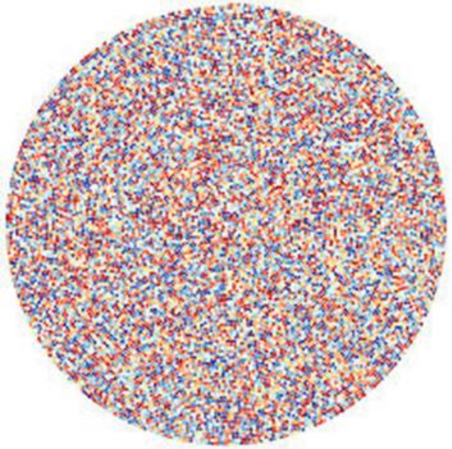
File Edit Shell Debug Options Window

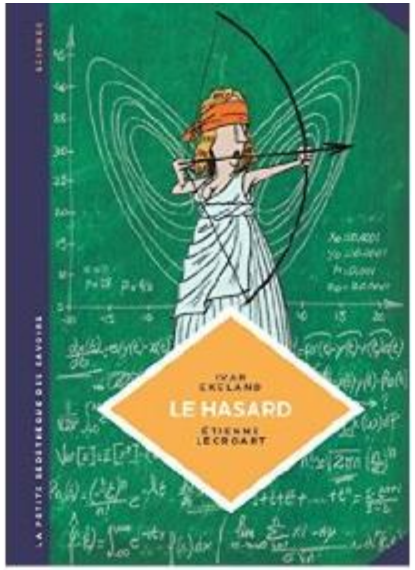
```
Python 3.5.2 (v3.5.2:4def2a2901a5
D64)] on win32
Type "copyright", "credits" or "l
>>>
 RESTART: C:\Data\1718\dnl premiè
ulation of pi\Leibniz1.py
>>> Euler(10)
3.04936163598207
>>> Euler(100)
3.132076531809105
>>> Euler(1000)
3.140638056205994
>>> Euler(10000)
3.1414971639472147
>>> Euler(100000)
3.1415831043264 56
```

L’ère du hasard

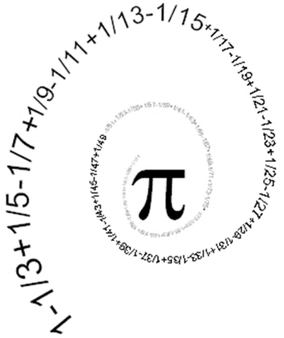


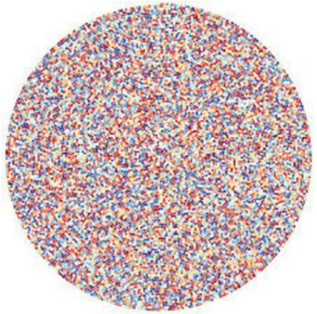


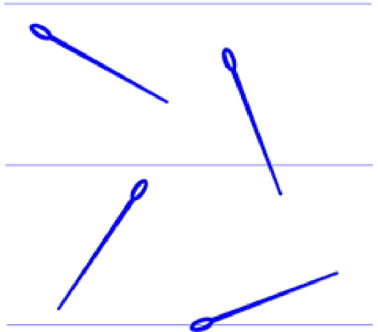


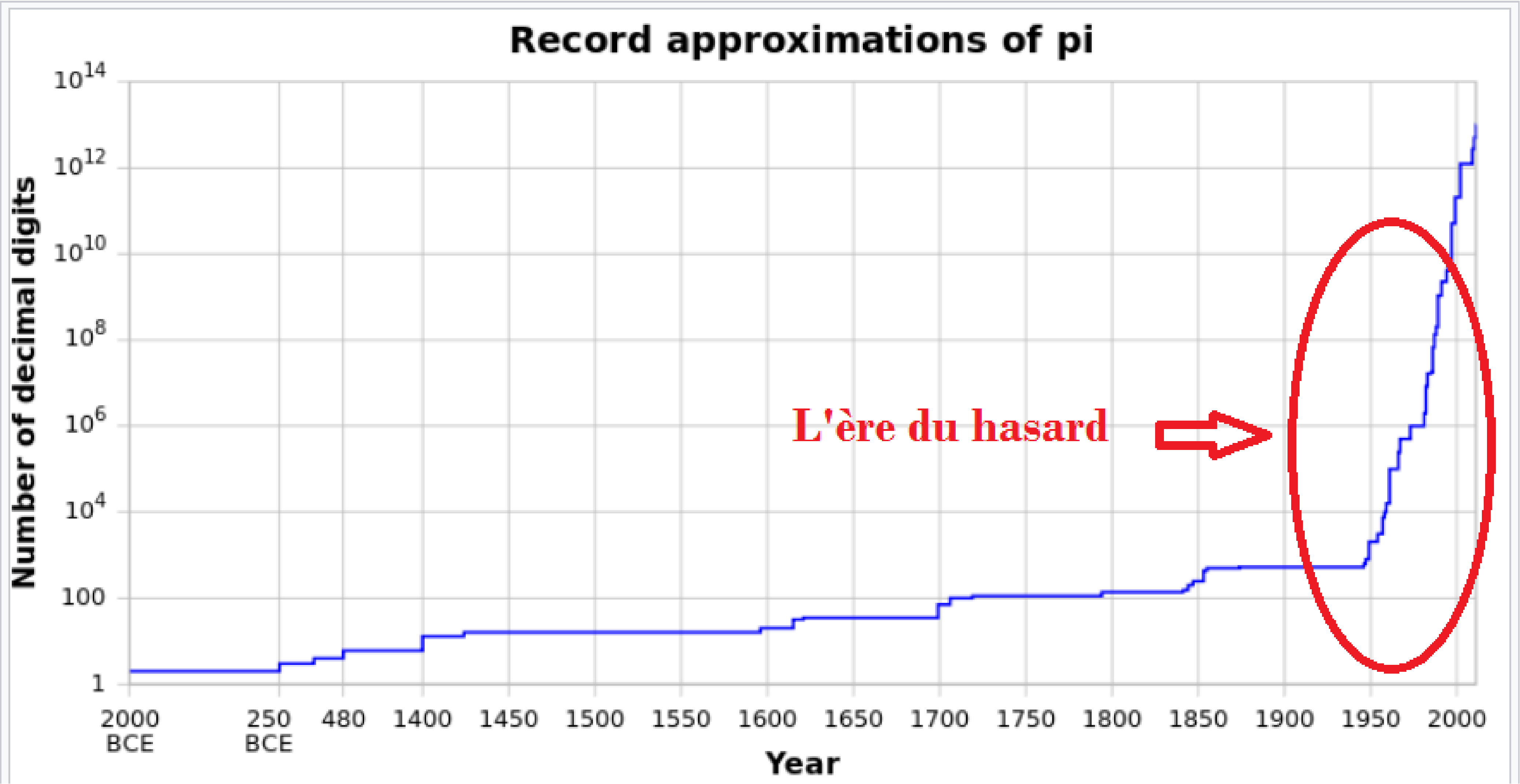


L’ère du hasard

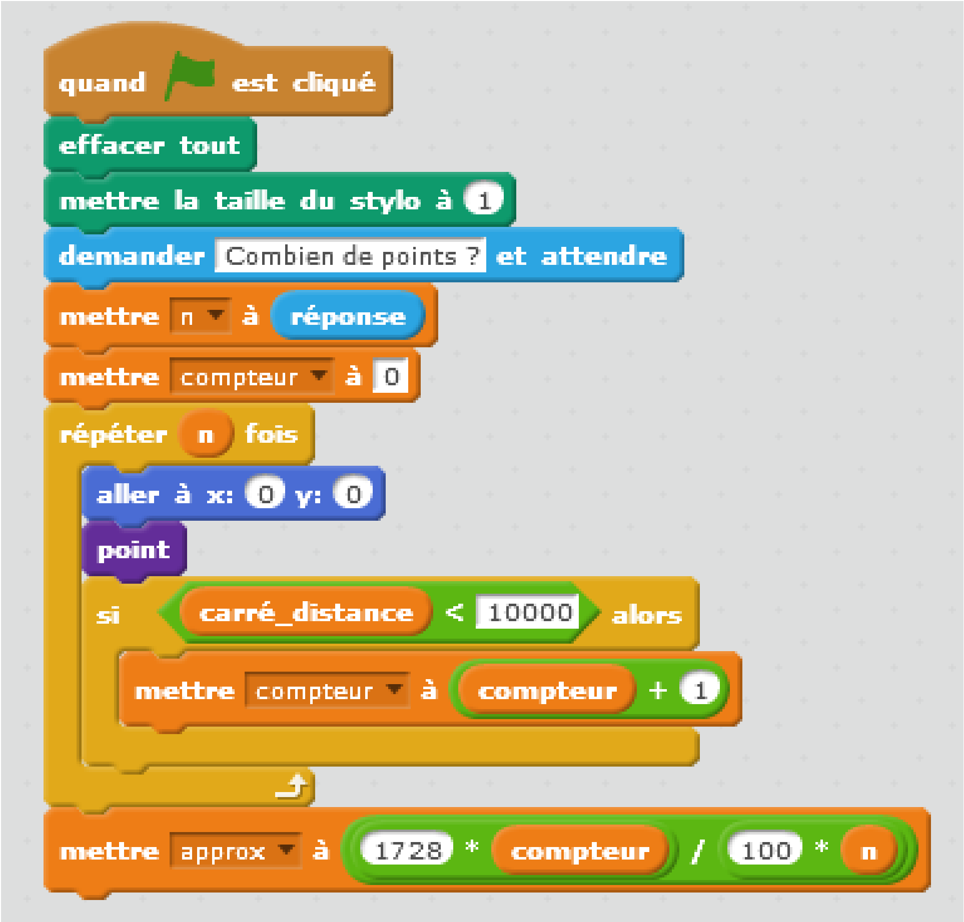


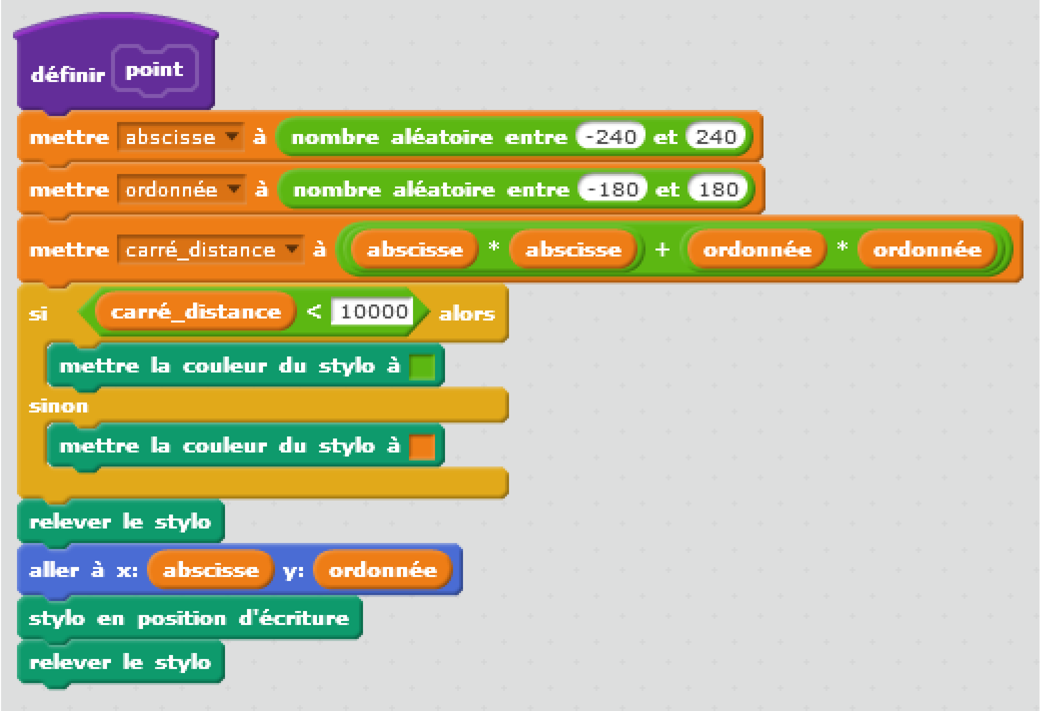




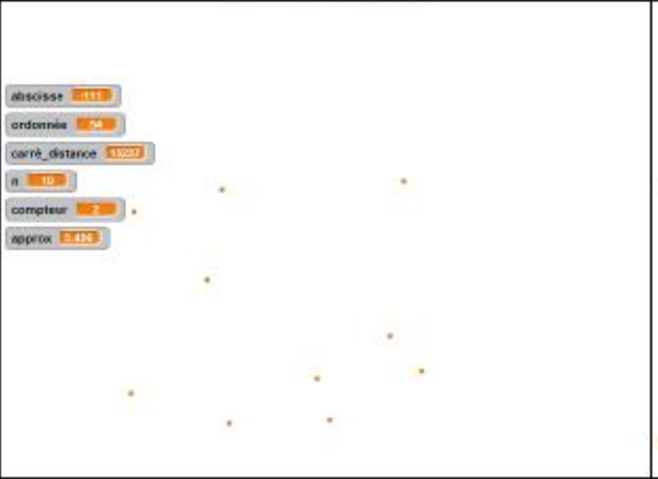
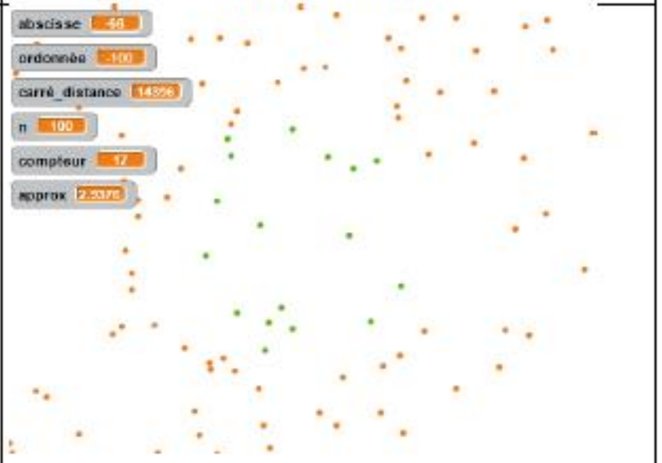
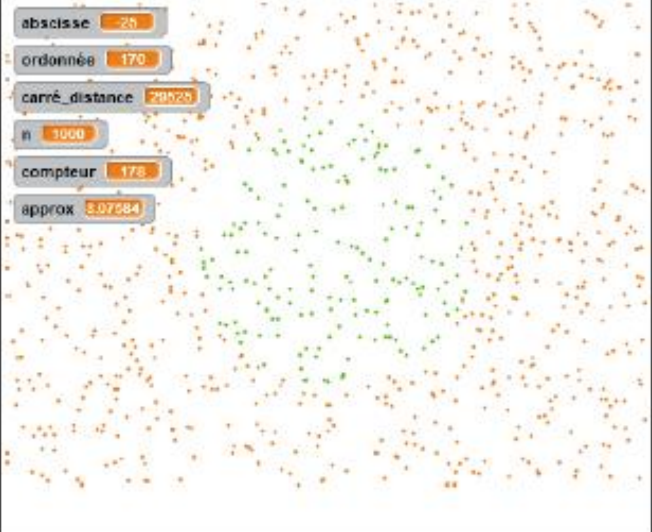
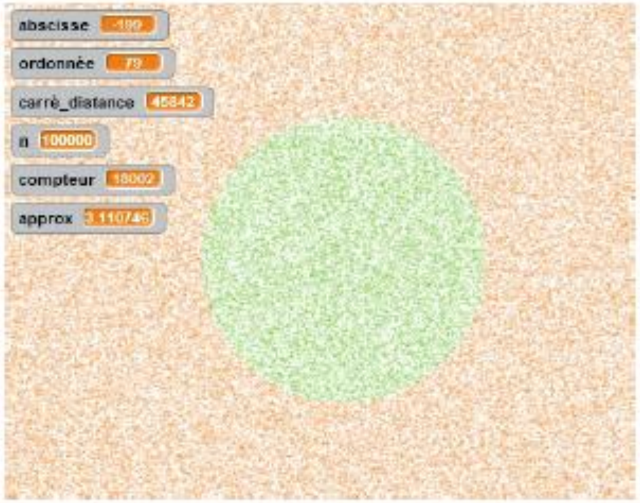


Autour de π	Monté Carlo sur Scratch



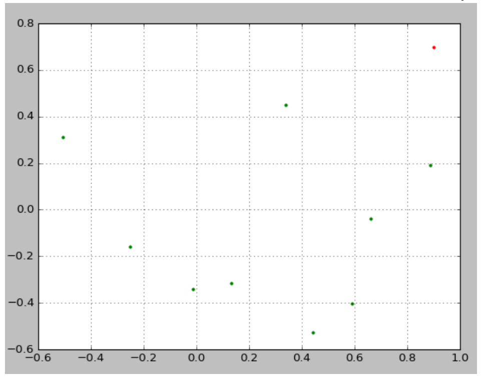


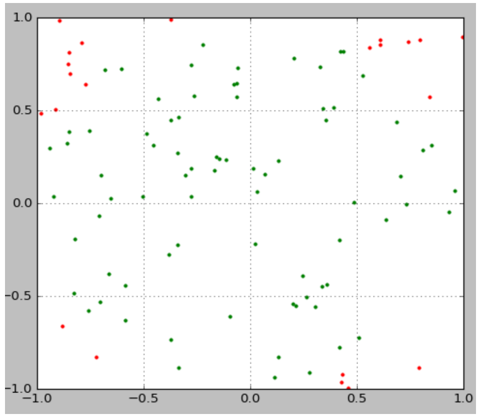
Autour de π	Monté Carlo sur Scratch

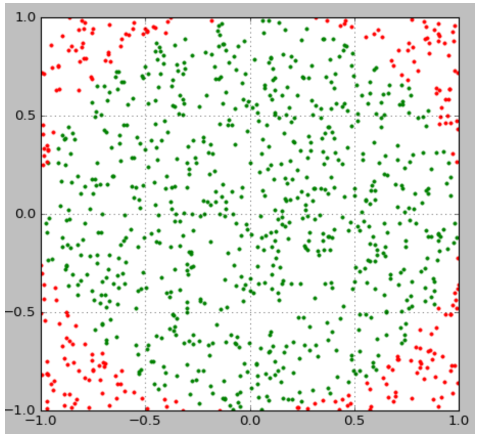
10 fléchettes	100 fléchettes
	
1 000 fléchettes	**100 000 fléchettes**
	

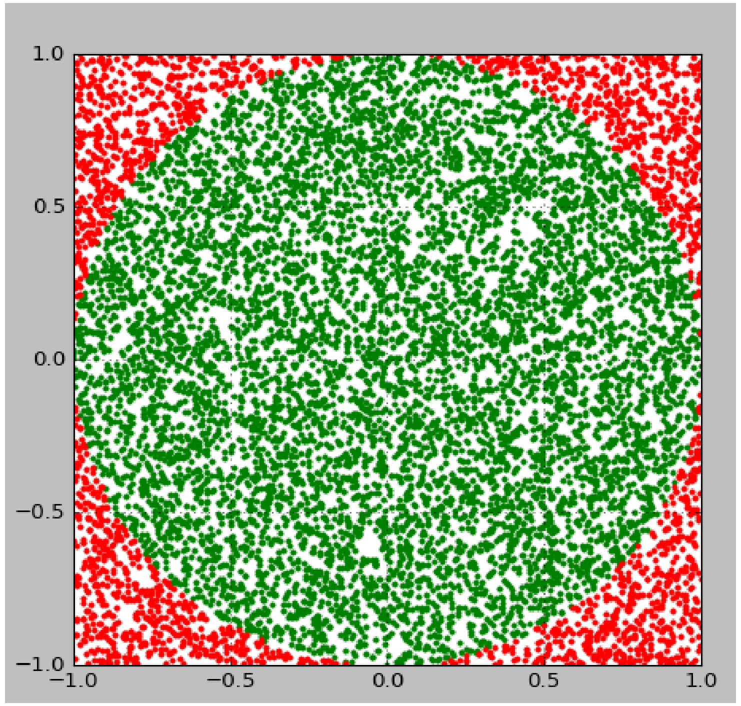
Autour de π	*Monté Carlo sur Python*

```python
from math import *
import random
import matplotlib.pyplot as plp
def monte_carlo(n):
    compteur=0
    for k in range(n):
        x=-1+2*random.random()
        y=-1+2*random.random()
        d=x**2+y**2
        if d<=1:
            compteur=compteur+1
    return(4*compteur/n)
def graph_mc(n):
    X_i=[]
    Y_i=[]
    X_e=[]
    Y_e=[]
    for k in range(n):
        x=-1+2*random.random()
        y=-1+2*random.random()
        d=x**2+y**2
        if d<=1:
            X_i.append(x)
            Y_i.append(y)
        else:
            X_e.append(x)
            Y_e.append(y)
    plp.plot(X_i,Y_i,'g.')
    plp.plot(X_e,Y_e,'r.')
    plp.grid()
    plp.show()
```










```
>>> monte_carlo(10)
2.8
>>> monte_carlo(100)
3.12
>>> monte_carlo(1000)
3.068
>>> monte_carlo(10000)
3.1404
>>> monte_carlo(100000)
3.13784
```

Georges Louis Leclerc de Buffon (1707-1788)

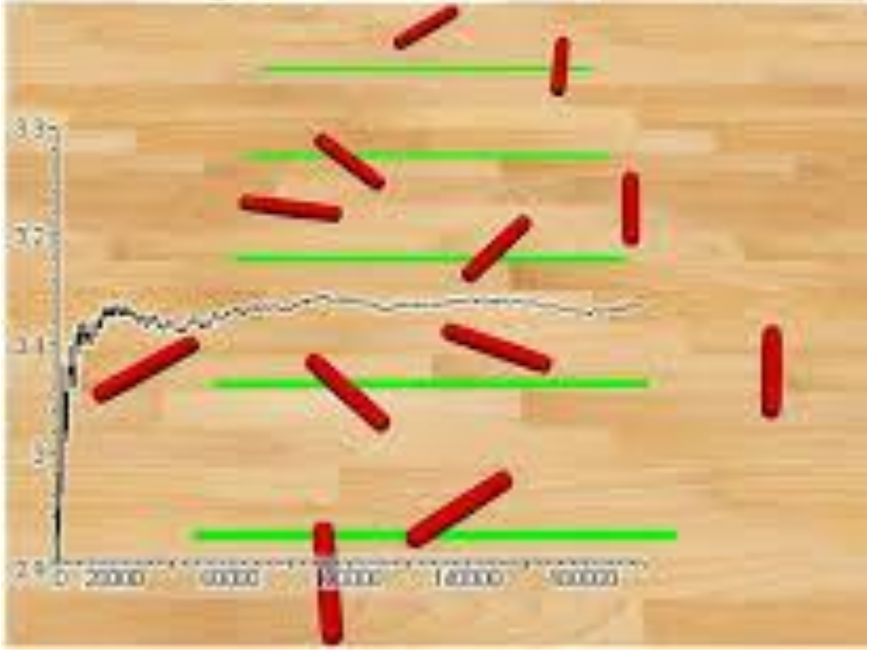
a montré que la probabilité qu'une aiguille de longueur L,

lancée sur un parquet dont les lattes ont une largeur L,

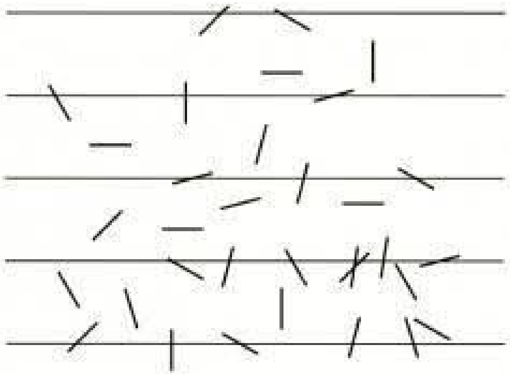
coupe le bord d'une latte est $\frac{2}{\pi}$.

C'est la première fois que la ***géométrie***

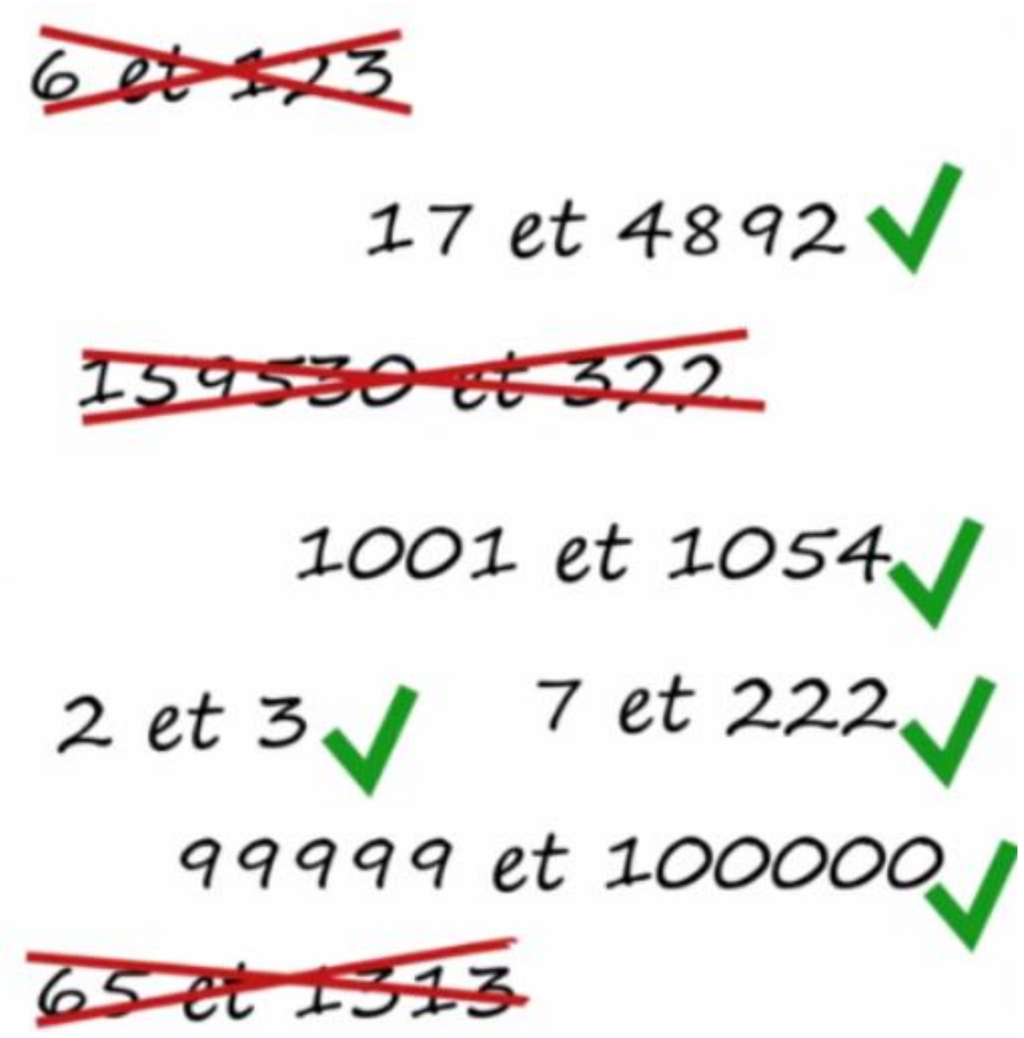
apparaît en termes en ***probabilités***.



$$
\begin{aligned}
P\left(\frac{X}{cos\Theta} < \frac{L}{2}\right) &= \int_{0<x<\frac{L}{2}\cos\theta} dx d\theta \\
&= \frac{4}{\pi D} \int_0^{\frac{\pi}{2}} \int_0^{\frac{L}{2}\cos\theta} dx d\theta \\
&= \frac{4}{\pi D} \int_0^{\frac{\pi}{2}} \frac{L}{2} cos\theta d\theta \\
&= \frac{2L}{\pi D}.
\end{aligned}
$$



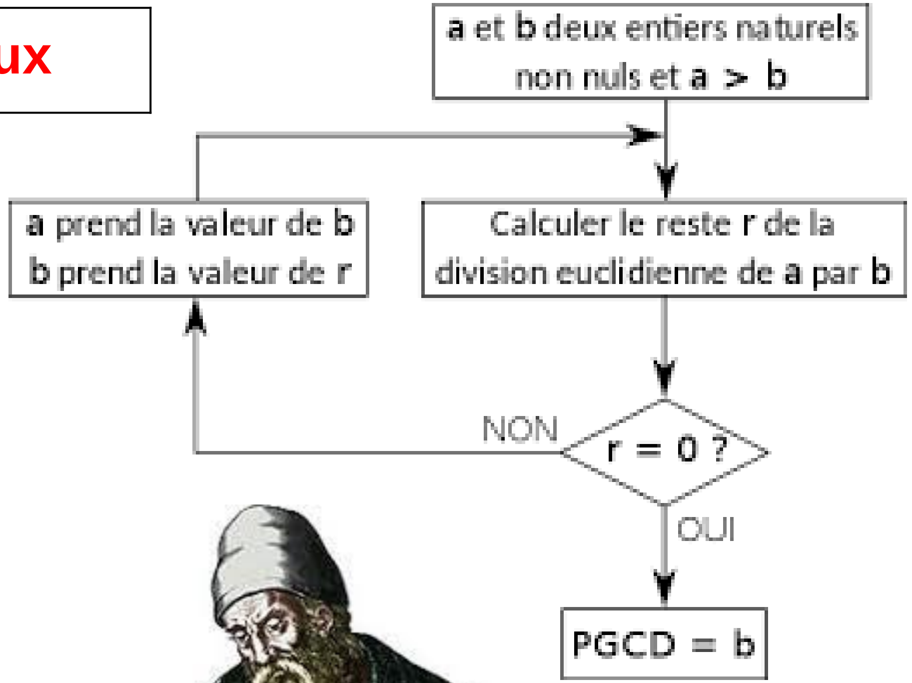
Autour de Π	Nombres premiers entre eux



Autour de π	**Nombres premiers entre eux**

Définition :
On dit que deux nombres ***a*** *et* ***b*** *sont* ***premiers entre eux*** *lorsque leur* ***Plus Grand Commun Diviseur*** *est égal à* ***1*** *; c'est à dire*

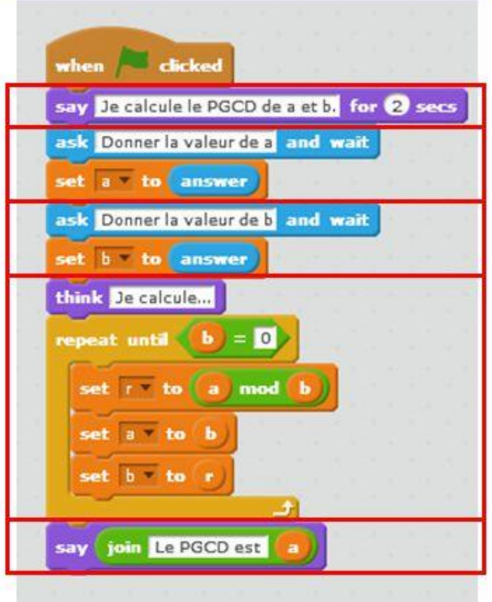
PGCD (a ; b) = 1

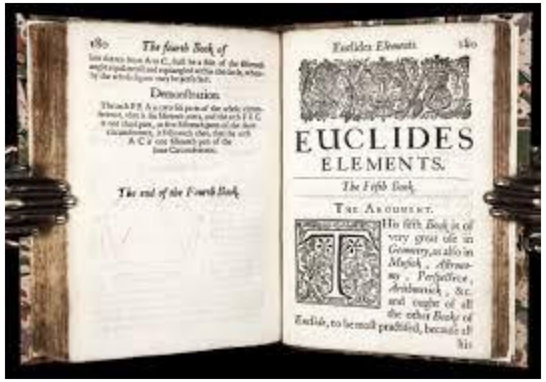


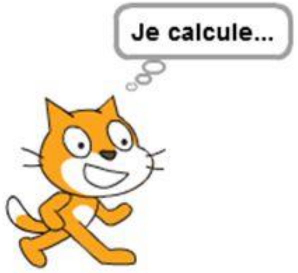
Algorithme d'Euclide – Calcul du PGCD de deux nombres entiers - **Sur SCRATCH**

Variables :
a est un nombre entier
b est un nombre entier
r est un nombre entier

Début algorithme
Lire a
Lire b
Tant que b>0
reste de a div b → r
b → a
r → b
Fin de Tant que
Afficher a
Fin algorithme







Autour de ∏	**Le théorème de Césaro**

Théorème *Cesàro*

Soient n un entier supérieur à 2 et $(a, b) \in \mathbb{N}^* \times \mathbb{N}^*$ tels que $a \leqslant b \leqslant n!$.

On note p_n la probabilité que a et b soient premiers entre eux.

Alors la suite (p_n) admet une limite et :

$$\lim_{n \to +\infty} p_n = \frac{6}{\pi^2}$$

Conclusion : la probabilité que deux entiers naturels non nuls, choisis au hasard, soient premiers entre eux est égale à $\frac{6}{\pi^2}$, soit environ 61 %.

Autour de Π | **Le théorème de Césaro**

Le théorème de Césaro

```
import random
def fprem(n):
  assert(n>=2)
  L=[]
  d=2
  while d*d<=n:
    while n%d==0:
     L.append(d)
     n=n//d
    d=d+1
  if n>1:
    L.append(n)
  return L
def f_prem_u(n):
    L=set(fprem(n))
    return(L)
def f_communs(n,k):
    L=f_prem_u(n)
    K=f_prem_u(k)
    I=L.intersection(K)
    return(I)
def nb_facteurs_communs(n,k):
    I=f_communs(n,k)
    a=len(I)
    return(a)
def cesaro(p):
    counter=0
    for i in range(p):
        n=random.randint(1,1000000)
        k=random.randint(1,1000000)
        a=nb_facteurs_communs(n,k)
        if a==0:
            counter=counter+1
    prop=counter/p
    return(prop)
```

Python 3.5.2 Shell

File Edit Shell Debug O

```
Python 3.5.2 (v3.5.2:
D64)] on win32
Type "copyright", "c
>>>
 RESTART: E:/1718/mat
>>> cesaro(100)
0.59
>>> cesaro(100000)
0.60736
>>>
```