TP- Cours sur les Intégrales en Python :

Application sur la méthode des rectangles et des trapèzes

Bloc de compétences	**Coder (IR)** **CONCEVOIR UNE STRUCTURE MATÉRIELLE ET LOGICIELLE (ER)** **3. Modéliser**	**C08** (IR) **C05(ER)** **C3 (Math)**
Logiciels utilisés	**Python avec l'interface Pyzo**	
Objectif pédagogique	Programmation Orientée Objet en Python	

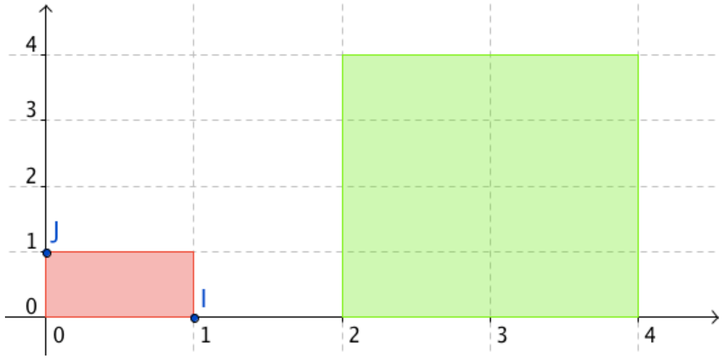
Connaissances issues du référentiel	Fonctions de référence Fonctions sinus et cosinus. Nombres complexes	
	savoir programmer une séquence, une instruction conditionnelle ou itérative comportant éventuellement un test d'arrêt.	C3 - Modéliser
	Langages de développement, de description, de création d'API et les IDE associés	Niveau 4 Niveau 3 Niveau 3

Extrait du cours de Yves Monka :

En 1696, Jacques Bernoulli reprend le mot latin « integer », déjà utilisé au XIVe siècle, pour désigner le calcul intégral. A cette époque, on partait de l'équation de la courbe pour calculer l'aire sous la courbe, c'est à dire du « bord » de la surface à la surface entière (intégrale).
Au milieu du XIXe siècle, les sciences sociales reprennent le mot pour exprimer l'idée qu'une personne s'intègre à un groupe.

Partie 1 : Intégrale et aire

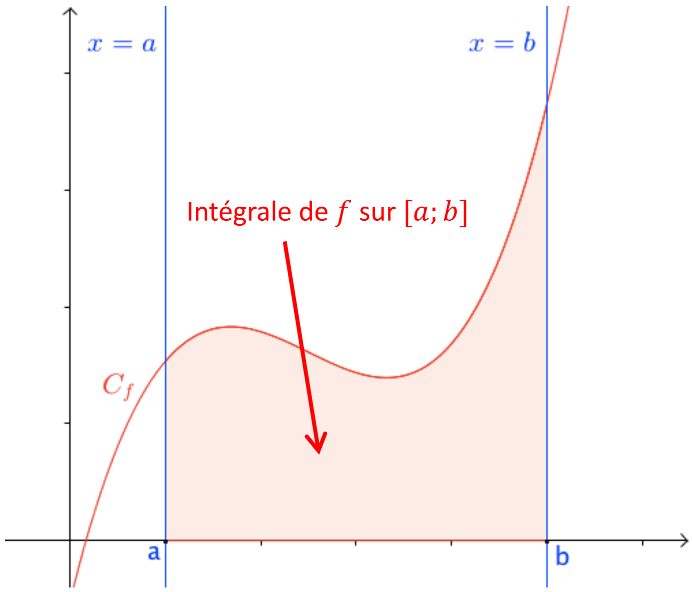
1) Unité d'aire



Dans le repère (O, I, J), le rectangle rouge a comme dimension 1 sur 1.
Il s'agit du rectangle "unité" qui a pour aire 1 unité d'aire. On écrit 1 u.a.
L'aire du rectangle vert est égale à 8 fois l'aire du rectangle rouge.
L'aire du rectangle vert est donc égale à 8 u.a.
Lorsque les longueurs unitaires sont connues, il est possible de convertir les unités d'aire en unités de mesure (le cm^2 par exemple).

2) Définition

<u>Définition :</u> Soit f une fonction continue et positive sur un intervalle $[a\,;b]$.
On appelle **intégrale** de f sur $[a\,;b]$ l'aire, exprimée en u.a., de la surface délimitée par la courbe représentative de la fonction f, l'axe des abscisses et les droites d'équations $x = a$ et $x = b$.



3) Notation

L'intégrale de la fonction f sur $[a\,;b]$ se note :

$$\int_a^b f(x)\,dx$$

Et on lit « intégrale de a à b de $f(x)\,dx$ ».

<u>Remarques :</u>
- a et b sont appelés les **bornes d'intégration**.
- x est la **variable d'intégration**. Elle peut être remplacée par toute autre lettre qui n'intervient pas par ailleurs.

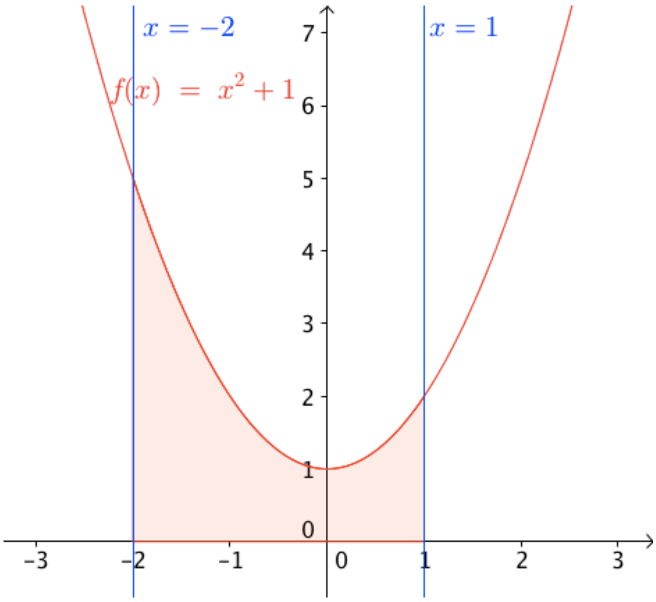
Ainsi on peut écrire : $\int_a^b f(x)\,dx = \int_a^b f(t)\,dt$.

"dx" ou "dt" nous permet de reconnaître la variable d'intégration.

Cette notation est due au mathématicien allemand *Gottfried Wilhelm von Leibniz* (1646 ; 1716). Ce symbole fait penser à un "S" allongé et s'explique par le fait que l'intégral est égal à une aire calculée comme somme infinie d'autres aires.
Plus tard, un second mathématicien allemand, *Bernhard Riemann* (1826 ; 1866) établit une théorie aboutie du calcul intégral.

Exemple :
L'aire de la surface délimitée par la courbe représentative de la fonction f définie par $f(x) = x^2 + 1$, l'axe des abscisses et les droites d'équations $x = -2$ et $x = 1$ est l'intégrale de la fonction f sur l'intervalle $[-2\,;\,1]$ et se note :

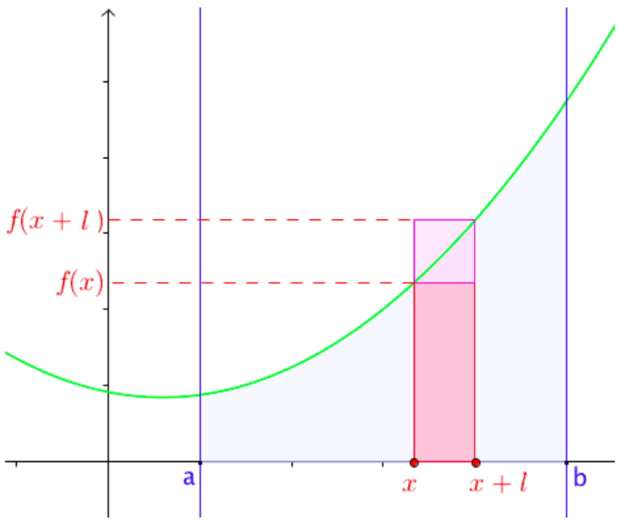
$$\int_{-2}^{1} x^2 + 1\,dx$$



4) Encadrement de l'intégrale d'une fonction monotone et positive

Soit une fonction f continue, positive et monotone sur un intervalle $[a\,;b]$.
On partage l'intervalle $[a\,;b]$ en n sous-intervalles de même amplitude $l = \frac{b-a}{n}$.
Sur un sous-intervalle $[x\,;x+l]$, l'aire sous la courbe est comprise entre l'aire de deux rectangles :

- l'un de dimension l et $f(x)$ qui a pour aire : $l \times f(x)$;
- l'autre de dimension l et $f(x+l)$ qui a pour aire $l \times f(x+l)$.



Sur l'intervalle $[a\,;b]$, l'aire sous la courbe est comprise entre la somme des n rectangles "inférieurs" et la somme des n rectangles "supérieurs".

Voici un algorithme écrit en langage naturel permettant d'obtenir un tel encadrement :

Langage naturel

Définir fonction rectangle(a, b, n)
 	
L ← (b-a)/n
x ← a
sommeInf ← 0
sommeSup ← 0
 	
Pour i allant de 0 à n-1
    sommeInf ← sommeInf +L f(*x*)
    x ← *x*+L
    sommeSup ← sommeSup + L f(*x*)
FinPour
 	
Afficher sommeInf et sommeSup

<u>Exemple :</u>
Avec Python, on programme cet algorithme pour la fonction $f(x) = x^2$ sur l'intervalle [1 ; 2].

```python
def f(x):
    return x**2

def rectangle(a,b,n):
    l = (b-a)/n
    x = a
    sommeInf = 0
    sommeSup = 0
    for i in range(n) :
        sommeInf = sommeInf + l*f(x)
        sommeSup = sommeSup + l*f(x + l)
        x = x + l
    return sommeInf,sommeSup
```

On exécute plusieurs fois le programme pour obtenir un encadrement de l'intégrale de la fonction carré sur [1 ; 2].

En augmentant le nombre de sous-intervalles, la précision du calcul s'améliore car l'encadrement formé de rectangles inférieurs et supérieurs se resserre autour de la courbe.

```
>>> rectangle(1,2,10)
(2.1850000000000014, 2.4850000000000017)
>>> rectangle(1,2,50)
(2.3034000000000017, 2.3634000000000017)
>>> rectangle(1,2,100)
(2.318350000000003, 2.3483500000000026)
>>>
```

On constate que plus le nombre *n* de rectangles est important, plus le calcul d'aire sera précis et ainsi plus la différence entre la somme des n rectangles "inférieurs" et la somme des n rectangles "supérieurs" sera faible.
Pour mieux mettre en évidence cette tendance , on peut tracer une courbe en utilisant la librairie *matplotlib*. Cela donne le code suivant :

```
from matplotlib.pyplot import plot,title,grid,legend,xlim,ylim,show
```

On importe les fonctions suivantes de la librairie malplotlib : elles permettent de tracer des courbes.

```
def f(x):
    return x**2

def rectangle(a,b,n):
    l = (b-a)/n
    x = a
    sommeInf = 0
    sommeSup = 0
    for i in range(n) :
        sommeInf = sommeInf + l*f(x)
        sommeSup = sommeSup + l*f(x + l)
        x = x + l
    return sommeInf,sommeSup
```

On retrouve la fonction écrite juste avant

On crée une liste qui contient les différents n pour lesquels on exécutera la fonction *rectangle()*

```
# Programme principal
listeN = [10,20,50,100,200,500]
Yinf = []
Ysup = []
```

On crée 2 listes dans lesquelles seront mémorisées les retours de la fonction *rectangle()*

```
for n in listeN :
    sommeInf,sommeSup = rectangle(1,2,n)
    Yinf.append(sommeInf)
    Ysup.append(sommeSup)
```

On exécute la fonction *rectangle()* pour les différentes valeurs de n

```
plot(listeN,Yinf,"b:x",label="Yinf")
plot(listeN,Ysup,"r:x",label="Ysup")
legend()
grid()
show()
```

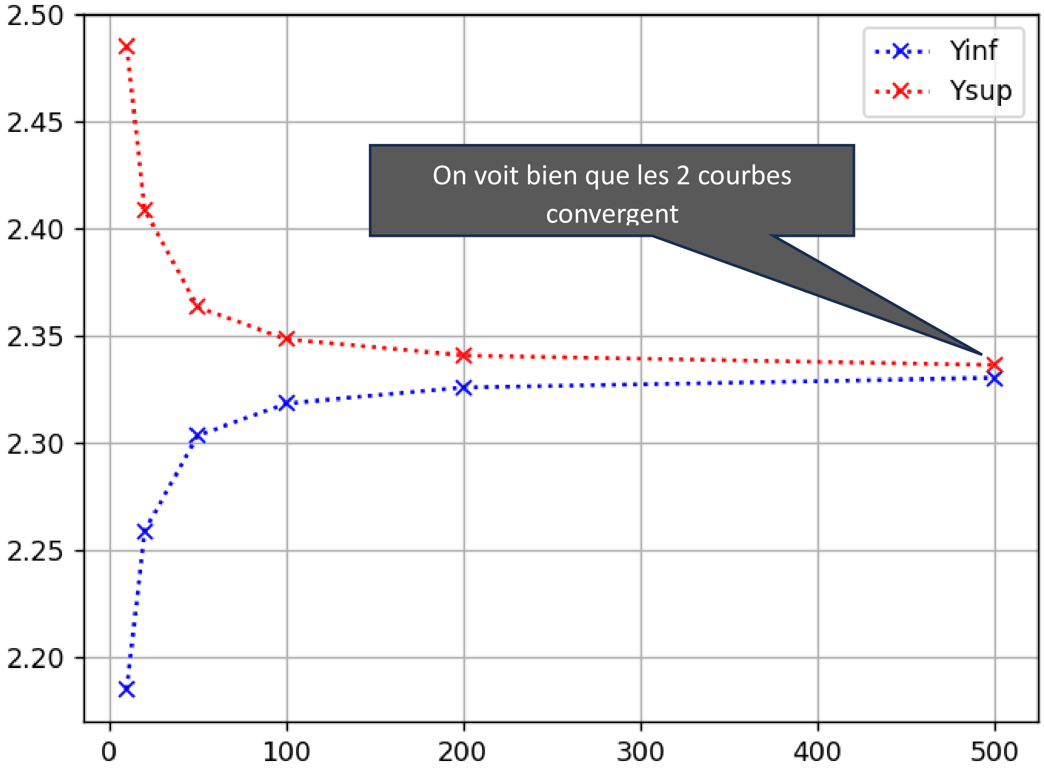
On exécute la fonction *plot()* 2 fois car on trace 2 courbes

On exécute la fonction *show()* pour finaliser le tracé des 2 courbes

⇨ Exécute le programme. Ferme la fenêtre et visualise le contenu des listes en exécutant dans la console :

```
>>> listeN
>>> Yinf
>>> Ysup
```

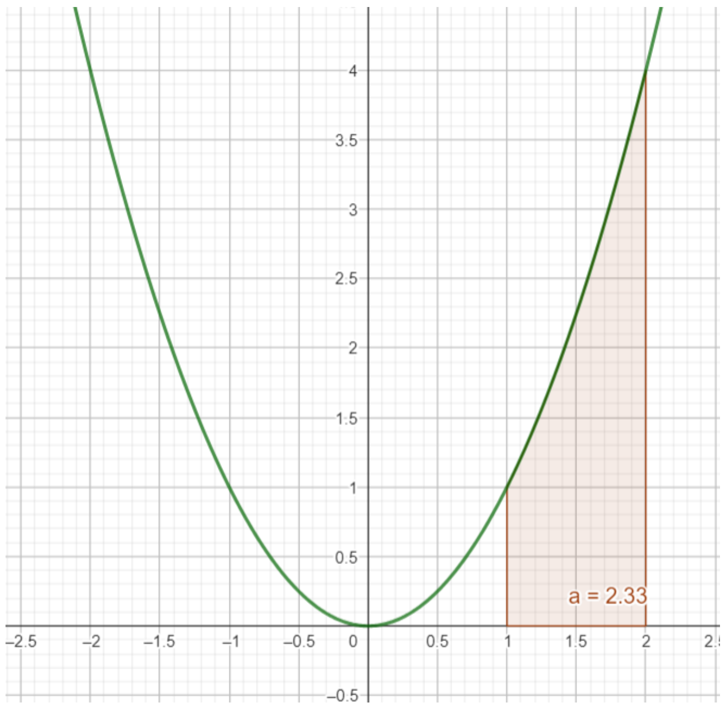
On obtient les 2 courbes suivantes :



Pour cet exemple avec la fonction $f(x) = x^2$ sur l'intervalle [1 ; 2], on obtient :

$$\int_1^2 x^2\,dx \approx 2{,}33$$

On peut confirmer ce résultat en ouvrant le logiciel Géogébra :



⇨ On exécute dans la ligne de saisie : $f(x) = x^2$
La courbe représentative de cette fonction s'affiche alors dans la zone graphique :

⇨ On exécute ensuite, toujours dans la ligne de saisie : a = Intégrale(f, 1, 2)

Le domaine entre la courbe et l'axe des abscisses, pour $1 < x < 2$ est alors coloré sur la fenêtre graphique. Géogébra donne également la valeur exacte de l'intégrale :

$$\int_1^2 x^2\,dx \approx 2{,}33$$

On retrouve ainsi la valeur obtenue avec notre code python précédent, pour des valeurs de n importantes.

<u>Travail à faire :</u>

1- Pour les 3 fonctions définies ci-dessous sur un intervalle particulier :

- Faire tourner l'algorithme précédent et donner la copie d'écran des courbes Yinf et Ysup, pour $n \in \{10\,;20\,;50\,;100\,;200\,;500\}$

- Tracer la courbe représentative sur Géogébra pour l'intervalle indiqué. Calculer l'intégrale exacte et comparer avec les valeurs retournées par la fonction *rectangle()*. Donner la copie d'écran de Géogébra.

   a. $f(t) = 0{,}1\,t^2 - 2$ *sur [-6 ;6]*

   b. $f(t) = \ t^3 + 1$ *sur [-1 ;1]*

   c. $f(t) = \ t^2 - 3t$ *sur [0 ;3]*

2- Conclure sur l'influence de la valeur de n sur le calcul de l'intégrale.

3- S'il vous reste du temps, coder une fonction nommée *trapeze()* qui permet de calculer l'intégrale en utilisant non pas la méthode des rectangles, mais celle dite « *des trapèzes* ». Cette nouvelle méthode est présentée ci-dessous.

4- Pour comparer la précision de ces différentes méthodes, ajouter sur les 2 courbes obtenues avec *plot(),* une troisième, relative à cette fonction *trapeze().*

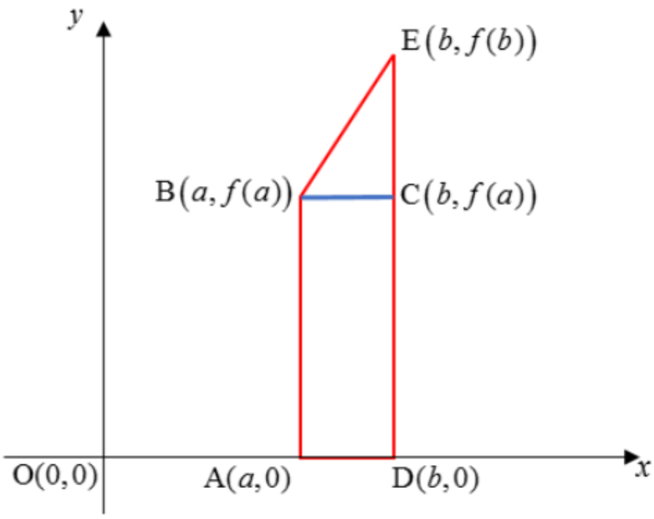
Méthode des trapèzes

Comme son nom l'indique, cette méthode d'intégration utilise une somme de surfaces de trapèzes.

Sur chaque intervalle, on réalise alors l'approximation suivante :

$$\int_a^b f(x)dx \approx \frac{b-a}{2}[f(a)+f(b)]$$

Justification de la formule



Pour calculer la surface du trapèze ABED, on fait la somme des aires du rectangle ABCD et du triangle rectangle BEC.

$$\text{surface du rectangle ABCD} = \text{AD} \times \text{AB} = (b-a)f(a)$$

$$\text{surface du triangle rectangle BEC} = \frac{\text{BC} \times \text{CE}}{2} = \frac{(b-a)\,[f(b)-f(a)]}{2}$$

$$\text{surface du trapèze ABED} = \frac{b-a}{2}[2f(a)+f(b)-f(a)] = \frac{b-a}{2}[f(a)+f(b)]$$