Sujet TP3 OL2

Objectifs pédagogiques : Calculer de manière approchée des intégrales avec les logiciels Scilab et Xcas.

Pour calculer une intégrale, un logiciel de calcul formel comme Xcas, recherche une primitive, ici avec la fonction `int` (ou `integrate` ou `integrer`) puis l'évalue entre les bornes, afin d'obtenir une valeur exacte.
Dans certains cas, il est inutile de calculer une primitive, soit parce qu'il n'en existe pas qui s'exprime par des fonctions élémentaires, soit parce qu'un calcul numérique est plus adapté (si le temps de calcul de la primitive est trop long, si la fonction présente des singularités dans l'intervalle d'intégration,..). Dans ce cas, on peut demander une valeur approchée en utilisant la fonction `evalf(int(...))` qui utilise une méthode approchée de calcul d'intégrale, basée sur la méthode des trapèzes.

1. Méthodes de calculs approchées

L'objectif est dans un premier temps de déterminer une valeur approchée de l'intégrale $I=\int_0^1 \sqrt{1-t^2}\,dt$ par les 3 méthodes vues en cours (voir poly Ma2) : les rectangles, les trapèzes et Simpson. Dans un deuxième temps, on comparera ces méthodes.

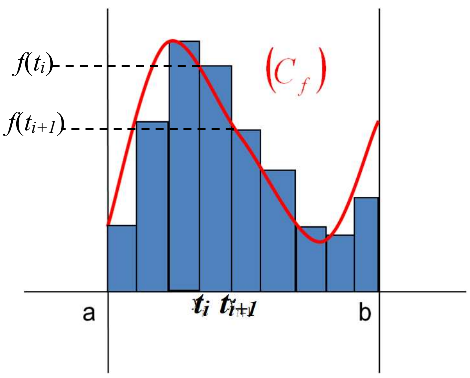
➢ **Exercice 1** : **avec xcas**

a. Tracer la fonction $f(t)=\sqrt{1-t^2}$ sur l'intervalle $[-1;1]$.

b. Déterminer la valeur exacte de $I=\int_0^1 \sqrt{1-t^2}\,dt$ avec **xcas** (ou « à la main » avec un changement de variable en posant $t=\sin x$).

c. Donner une valeur approchée, avec xcas, de $I=\int_0^1 \sqrt{1-t^2}\,dt$ à **10^{-4} près.**

➢ **Exercice 2** : **Méthode des rectangles à gauche**



On prend une subdivision de l'intervalle [a,b] à pas constant :
$dt=(t_{i+1}-t_i)=\dfrac{b-a}{N}$ où ***N* est le nombre d'intervalles entre *a* et *b*.**

Sur l'intervalle $[t_i\,;t_{i+1}]$ on approxime l'intégrale $\int_{t_i}^{t_{i+1}} f(t)\,dt$ par **l'aire du rectangle** : $\text{Hauteur}\times\text{Base}=f(t_i)\cdot(t_{i+1}-t_i)=f(t_i)\cdot\dfrac{b-a}{N}$

On a alors la valeur de l'intégrale $\int_a^b f(t)\,dt$ qui est approchée par la **somme des rectangles** : $\boxed{I\simeq R_N(f)=\dfrac{b-a}{N}\cdot\sum_{i=0}^{N-1} f(t_i)}$

a. Proposer un algorithme descriptif qui calcul la valeur approchée d'une intégrale $I=\int_a^b f(t)\,dt$ avec la méthode de la somme des rectangles.

b. Programmer sous scilab **l'algorithme de la méthode des rectangles** pour déterminer une valeur approchée de $I=\int_0^1 \sqrt{1-t^2}\,dt$ pour une valeur d'intervalles N donnée.

i. en utilisant une boucle for
ii. en optimisant avec la fonction `sum` de scilab

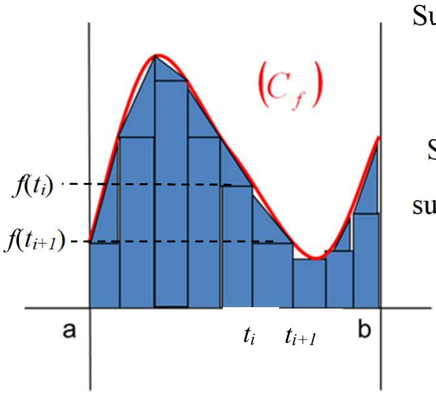
c. Mettre dans votre CR les programmes et un exemple d'exécution pour N=10.

d. Relever les valeurs approchées de I à **10^{-3} près** pour N = 10, 20,50, 100, 200, 300.
Dans votre compte-rendu, faire une synthèse de vos résultats dans un tableau comme celui-ci :

N	**10**	**20**	**50**	**100**	**200**	**300**
I Rectangle						

☞ *Indication : dans votre fichier scripte « .sce » on pourra se mettre* en `format(7)` pour avoir moins de chiffres décimaux.

➢ <u>Exercice 3</u> : Méthode des trapèzes



Sur l'intervalle $[t_i\,;t_{i+1}]$ on approxime $\int_{t_i}^{t_{i+1}} f(t)\,dt$

par **l'aire du trapèze** : $\dfrac{\left(f(t_{i+1})+f(t_i)\right)\cdot\left(t_{i+1}-t_i\right)}{2}$.

Si N est le nombre d'intervalles entre a et b alors on prend une subdivision à pas constant : $dt=(t_{i+1}-t_i)=\dfrac{b-a}{N}$.

On a alors l'intégrale $\int_a^b f(t)\,dt$ qui est approchée par la

somme des trapèzes : $\boxed{T_n(f)=\dfrac{b-a}{N}\cdot\sum_{i=0}^{N-1}\left(\dfrac{f(t_i)+f(t_{i+1})}{2}\right)}$

a. Proposer un algorithme descriptif de la méthode des trapèzes.

b. Programmer sous scilab **l'algorithme de la méthode des trapèzes** pour déterminer une valeur approchée de $I=\int_0^1\sqrt{1-t^2}\,dt$ pour une valeur d'intervalles N donnée.

i. en utilisant une boucle for
ii. en optimisant avec la fonction `sum` de scilab

c. Mettre dans votre CR les programmes et un exemple d'exécution pour N=10.

d. Relever les valeurs approchées de I à 10^{-3} près pour N = 10, 20, 50, 100, 200, 300.
Dans votre compte-rendu, faire une synthèse de vos résultats dans un tableau.

➢ <u>Exercice 4</u> : Méthode de Simpson

Sur l'intervalle $[t_i\,;t_{i+1}]$ on approxime $f(t)$ par la fonction polynôme **de degré inférieur ou égale à 2,** coïncidant avec f en $t_i\,;\dfrac{t_i+t_{i+1}}{2}$ et t_{i+1}. On montre alors l'intégrale $\int_a^b f(t)\,dt$ est approchée par la somme : $\boxed{S_n(f)=\dfrac{b-a}{6N}\cdot\sum_{i=0}^{N-1}\left(f(t_i)+f(t_{i+1})+4f\left(\dfrac{t_i+t_{i+1}}{2}\right)\right)}$

a. Proposer un algorithme descriptif de la méthode de Simpson.

b. Programmer sous scilab **l'algorithme de la méthode de Simpson** pour déterminer une valeur approchée de $I=\int_0^1\sqrt{1-t^2}\,dt$ pour une valeur d'intervalles N donnée.

c. Mettre dans votre CR les programmes et un exemple d'exécution pour N=10.

d. Relever les valeurs approchées de I **à 10^{-3} près** pour N = 10, 20, 50, 100, 200, 300.
Dans votre compte-rendu, faire une synthèse vos résultats dans un tableau.

- **<u>Exercice 5</u>** : **Comparaison des méthodes**

  a. Pour la méthode des rectangles faire un programme qui donne le **nombre minimal** N d'intervalles nécessaires pour donner une valeur approchée de I **à 10^{-3} près.**
  (on pourra utiliser la **valeur exacte de** I obtenue dans **l'exercice 1** comme valeur de référence)

  b. Idem avec les méthodes des trapèzes et de Simpson.

  c. Conclure.