Exercice 1 (Méthode d'Euler appliquée à une EDL1)

La méthode d'Euler est un algorithme itératif permettant d'approcher la solution d'une équation différentielle à l'aide de développements limités.

On considère l'équation différentielle linéaire $(E) : xy'(x) + 2y(x) = \dfrac{x^2}{1+x^2}$

Dans les calculs de résolution, on pourra utiliser le fait que $x^3 = x(1+x^2) - x$.

1. Soit y une solution de (E), prenant la valeur y_0 au point x_0.

   On note $F(x,y) = \dfrac{x}{1+x^2} - 2\dfrac{y}{x}$ et h un réel positif “assez petit”.

   (a) Expliquer pourquoi on peut faire l'approximation :
   $$y(x_0+h) \simeq y_0 + h \times F(x_0, y_0)$$
   On note y_1 cette approximation : $y_1 = y_0 + h \times F(x_0, y_0)$.

   (b) En notant $x_i = x_0 + i \times h$, quelle approximation donner pour $y(x_2)$?

   (c) On construit ainsi par récurrence la suite des valeurs y_i :
   $y_{i+1} = y_i + hF(x_i, y_i)$.
   En s'appuyant sur la question précédente, on montre facilement par récurrence que pour tout i, $y(x_i) \simeq y_i$.
   Écrire une fonction Python `F` de paramètres x et y qui renvoie la valeur de $F(x,y)$.
   Écrire une fonction Python `euler` qui renvoie, à partir de x_0, y_0, le pas h, et la borne supérieure b de l'intervalle de résolution, les deux listes $[x_i]$ et $[y_i]$.

2. On admettra que la solution de (E) vérifiant la condition initiale $y(1) = \frac{1}{2}$ est la fonction f définie par $f(x) = \dfrac{x^2 - \ln\left(x^2+1\right) + \ln(2)}{2x^2}$.
   Écrire une fonction python `comparaison` d'arguments `h, b` qui trace sur une même figure la courbe de f et la ligne brisée des points de coordonnées (x_i, y_i). On choisira $h = 0.1$, $b = 10$, $x_0 = 1$ et $y_0 = \frac{1}{2}$.

3. À quel niveau y a-t-il décalage entre les 2 courbes ? pourquoi ?

Exercice 2 (Méthode d'Euler pour une équation différentielle non linéaire)
Soit l'équation différentielle $(1+x)y' - \cos(y) = x$.
En vous inspirant de l'exercice précédent, écrire une fonction Python `euler2` qui renvoie, à partir de x_0, y_0, le pas h, et la borne supérieure b de l'intervalle de résolution, les deux listes $[x_i]$ et $[y_i]$ obtenues par la méthode d'Euler.

Représenter graphiquement quelques solutions de cette équation différentielle. On ne cherchera pas à la résoudre de façon exacte, on se contentera de la résolution numérique que constitue la méthode d'Euler. Pour l'illustration on pourra utiliser `euler2` avec les deux jeux de paramètres $(-0.7, 1, 0.01, 5)$ et $(-5, 0, 0.01, -1.3)$.

Exercice 3 (Méthode de Newton : approximation d'une solution de $f(x) = 0$)

La méthode de Newton est un algorithme itératif permettant d'approcher une solution de l'équation $f(x) = 0$ où f est une fonction de classe C^1 vérifiant certaines conditions.

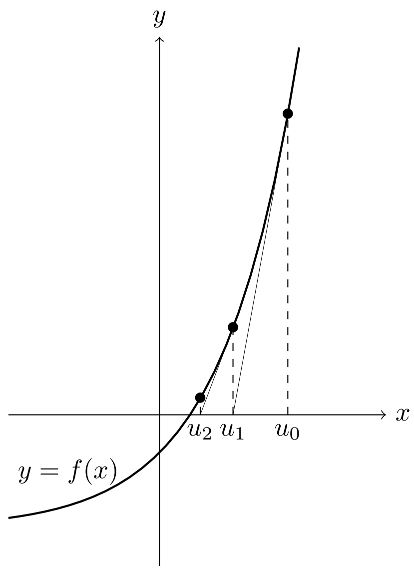
Préliminaires : présentation de la méthode de Newton

Soit f une fonction de classe C^1 sur un intervalle $[a,b]$. On suppose que $f(a)f(b) < 0$ et que f' ne s'annule pas sur $[a,b]$.
Soit $u_0 \in [a,b]$.
On suppose définis les termes $u_0, \ldots, u_n$ et on va définir u_{n+1} de la façon suivante :
Soit $\mathscr{T}_n$ la tangente à $\mathscr{C}_f$ au point u_n. Le nombre u_{n+1} est l'abscisse du point d'intersection de $\mathscr{T}_n$ avec l'axe des abscisses.

On définit ainsi par récurrence une suite $(u_n)_{n\in\mathbb{N}}$ lorsque f et u_0 vérifient certaines conditions que l'on n'exposera pas dans ce document.

On dit que la suite $(u_n)_{n\in\mathbb{N}}$ est obtenue à partir de la fonction f et du premier terme u_0 par la méthode de Newton.

Illustration de la méthode de Newton



1. **Convergence de la suite $(u_n)_{n\in\mathbb{N}}$**
   (a) Déterminer la relation de récurrence liant u_{n+1} et u_n.
   On supposera que la suite $(u_n)_{n\in\mathbb{N}}$ est bien définie et que $\forall n \in \mathbb{N}$, $u_n \in [a, b]$.
   (b) Montrer que si la suite (u_n) est bien définie et converge vers ℓ, alors $f(\ell) = 0$.

2. **Application à la fonction $x \longmapsto x^3 - \alpha$**
   (a) **Étude mathématique**

   Soit $\alpha > 1$. On considère la fonction $f : \begin{cases} [1,\alpha] & \longrightarrow & \mathbb{R} \\ x & \longmapsto & x^3 - \alpha \end{cases}$.

   i. Montrer que f vérifie les conditions des préliminaires et ne s'annule qu'en un seul point que l'on notera β.
   ii. Montrer que l'application de la méthode de Newton conduit à la relation de récurrence $u_{n+1} = \varphi(u_n)$ pour tout $n \in \mathbb{N}$, où $\varphi(x) = \frac{2x}{3} + \frac{\alpha}{3x^2}$. On posera par ailleurs $u_0 = \alpha$.
   iii. Étudier le signe de $\varphi(x) - x$ sur $[\beta, \alpha]$, puis les variations de φ sur $[\beta, \alpha]$. En déduire que $\forall x \in [\beta, \alpha]$, on a $\varphi(x) \in [\beta, \alpha]$.
   iv. Déduire de la question précédente que :
      A. (u_n) est bien définie et à valeurs dans $[\beta, \alpha]$.
      B. (u_n) est décroissante.
   v. Conclure quant à la convergence et la limite de (u_n).

   (b) **Codage de l'algorithme en langage Python**

   Il s'agit maintenant d'implémenter la méthode de Newton en langage Python.

   i. Écrire une fonction `newton` ayant comme paramètres `alpha` et `précision`, qui renvoie une valeur approchée de $\sqrt[3]{\texttt{alpha}}$ à la précision `precision` souhaitée, en appliquant la méthode de Newton.
   ii. À l'aide de cette méthode, déterminer une valeur approchée de $\sqrt[3]{2}$ à 10^{-15} près. Combien d'itérations ont-elles été nécessaires pour obtenir cette valeur approchée ? Comparer avec le nombre d'itérations nécessaires par la méthode de dichotomie (on ne réécrira pas l'algorithme de dichotomie, on se basera uniquement sur le fait que l'intervalle initial $[a, b]$ est réduit de moitié à chaque itération).
   iii. Écrire une fonction `resolution` qui prend en argument `f` , `u0`, `precision`, `maximum` et qui applique la méthode de Newton pour résoudre $f(x) = 0$, avec pour premier terme `u0`, jusqu'à ce que `precision` soit atteinte ou que le nombre d'itérations ait dépassé `maximum` (dans ce cas la fonction renverra un message avertissant l'utilisateur que la méthode a échoué).
   Pour les calculs de dérivées de la fonction `f` , on appliquera les résultats d'approximation numérique de dérivée vues en cours.

3. **Application à la fonction $x \longmapsto x^2 - \alpha$**
   Montrer que l'application de la méthode de Newton à la fonction $f : x \mapsto x^2 - \alpha$ correspond à la méthode de Héron ou algorithme de Babylone permettant d'extraire la racine carrée de α (voir exo 3 du TD 2 d'info).