DS 3

Informatique pour tous, première année

Julien Reichert

Exercice 1 : Écrire une fonction qui prend en entrée une liste de listes `LL` et une valeur quelconque et qui détermine la liste des positions en tant que couples (ligne, colonne) où l'élément figure dans `LL`. L'ordre n'est pas imposé.

Par exemple, la fonction retournera `[(0, 2), (1, 1)]` si elle est appelée pour `[[1, 2, 3], [2, 3]]` et `3`.

Exercice 2 : Écrire une fonction qui réalise la méthode d'Euler en prenant en entrée une fonction `G` représentant une équation différentielle d'ordre un, une condition initiale `y0` et au choix un tableau de valeurs `X` (à la manière du cours en MPSI 2) ou deux nombres `x0` et `xn` permettant de créer soi-même le tableau de valeurs `X` (à la manière du cours en MPSI 1). Si y est la solution de l'équation différentielle en question telle que $y($`X[0]`$) =$ `y0`, la fonction devra retourner un tableau `Y` de taille `len(X)` tel que `Y[0]` soit `y0` et pour tout autre indice `i` on ait `Y[i]` valeur approchée de $y($`X[i]`$)$.

Exercice 3 : Écrire une fonction qui réalise la méthode de Newton en prenant en entrée une fonction `f`, sa dérivée `df`, un point de départ `x0` et un seuil `eps`. La fonction devra retourner une approximation d'un antécédent de zéro par la fonction en question.

Exercice 4 : On considère une variante de la méthode de Newton dans laquelle on n'assimile pas localement la courbe de la fonction à une droite mais à une parabole. On cherche alors un point d'intersection de la parabole et de l'axe des abscisses (dans le cas où il y en a un...). Comme il existe une infinité de fonctions polynomiales de degré deux ou moins telles que $f(x_0) = a$ et $f'(x_0) = b$, on utilise une troisième information, qui peut s'appuyer sur un autre point ou la dérivée seconde, le choix que nous retiendrons ici.

Question 4.1 : Donner l'unique fonction polynomiale de degré ≤ 2 telle que $f(x_0) = a$, $f'(x_0) = b$ et $f''(x_0) = c$ en fonction de a, b, c et x_0.

Question 4.2 : Avec les mêmes paramètres, déterminer les valeurs x qui annulent cette fonction en distinguant les cas.

Question 4.3 : Écrire en Python une fonction qui détermine en fonction de ses paramètres `a`, `b`, `c` et `x0` la solution la plus proche du point `x0` à l'équation $f(x) = 0$ où f est la fonction trouvée précédemment, avec comportement au choix si f ne s'annule pas.

Question 4.4 : Écrire la variante de la méthode de Newton s'appuyant sur la question précédente pour trouver chaque point de l'approximation en fonction du précédent, sachant que dans les cas où la parabole ne croise pas l'axe des abscisses, on pourra utiliser la tangente comme dans la méthode de Newton de base.

Pour cette dernière question, le principe est donc de déterminer dans chaque tour de boucle le point d'intersection de la parabole déterminée à l'aide de la question 4.1 et de l'axe des abscisses, en se servant de la fonction écrite à la question 4.3, et si aucun tel point n'existe, on déterminera le point d'intersection de la tangente à la courbe en le point actuel et l'axe des abscisses, et si cette tangente est horizontale... tant pis, il y aura une erreur. La première ligne de la question sera obligatoirement :

```
def newton_parabole(f, df, ddf, x0, eps):
```

où `f` est la fonction dont on cherche un zéro, `df` est sa dérivée, `ddf` est sa dérivée seconde, `x0` est le point de départ et `eps` est le seuil utilisé pour arrêter la recherche.