

DS 3

Informatique pour tous, première année

Julien REICHERT

Exercice 1 : Écrire une fonction prenant en argument une fonction f , deux nombres a et b pour lesquels on suppose que $a < b$, ainsi qu'un entier n , et qui calcule une valeur approchée de l'intégrale de f entre a et b par une méthode vue en cours.

Exercice 2 : Rappeler la complexité de l'algorithme du pivot de Gauss dans sa version standard, et expliquer l'impact de l'utilisation du pivot partiel sur cette complexité.

Remarque : On donne ci-dessous une version de l'algorithme. Les lignes entre parenthèses peuvent être utilisées comme repères pour les exercices 2 et 3.

```
(1) def pivot(M):
(2)     n,m = len(M), len(M[0])
(3)     for i in range(n):
(4)         if M[i][i] == 0.:
(5)             j = i+1
(6)             while (j < n and M[j][i] == 0.):
(7)                 j += 1
(8)             if j <= n:
(9)                 raise ValueError("Le système n'est pas de Cramer.")
(10)            M[i], M[j] = M[j], M[i]
(11)            rapport = M[i][i]
(12)            for k in range(m):
(13)                M[i][k] = M[i][k] / rapport
(14)            for j in range(n):
(15)                if j != i:
(16)                    rapport = M[j][i]
(17)                    for k in range(m):
(18)                        M[j][k] = M[j][k] - rapport*M[i][k]
```

Exercice 3 : Une erreur a été volontairement ajoutée au code ci-dessus. Corriger la ligne où elle se trouve.

Exercice 4 : Écrire une fonction ayant pour arguments une liste de listes l , et qui renvoie la version « aplatie » de l , c'est-à-dire une liste dont les éléments sont les éléments des listes composant l . L'ordre n'est pas imposé.

Par exemple, l'appel de la fonction sur la liste $[[1,2], [3,4], [1], [1,2]]$ renverra $[1,2,3,4,1,1,2]$.

Exercice 5 : Écrire une fonction ayant pour arguments trois flottants deb , $seuil$ et pas et retournant le tableau $arange(deb,seuil,pas)$. On rappelle que $arange$ est similaire à $range$, mais accepte les flottants (et sa valeur de retour est un tableau, donc). La fonction à écrire doit renvoyer le résultat de l'appel de la fonction $linspace$ avec trois arguments : deb , fin et $nombre$; ce résultat est le tableau dont le premier élément est deb , le dernier est fin et les éléments forment le début d'une suite arithmétique. Bien entendu, l'utilisation des boucles et des fonctions $range$ et $arange$ est interdite.

Par exemple, l'appel à $pseudoarange(0,5,1)$ renverra $[0., 1., 2., 3., 4.]$ en tant que $linspace(0,4,5)$.