

TP6 : Classes, matrices creuses, Laplacien

On continue le TP6. Téléchargez les fichiers joints au TP dans un même répertoire. Ces fichiers contiennent une version du code actualisée contenant le constructeur de la matrice de la discrétisation du laplacien en 1D, et la fonction `matcreuse_vec` qui effectue une multiplication matrice creuse - vecteur.

1 Matrices creuses

1. Créer une nouvelle classe `solution`, qui contient comme données privées un entier `siz` et un vecteur de réels (type `vector`) de taille `siz`.
2. Créer pour la classe `solution` le constructeur par défaut.
3. Créer pour la classe `solution` un constructeur prenant en argument un entier et un réel, qui permet d'initialiser une instance de `solution` à une taille donnée et à une valeur constante. Dans ce deuxième constructeur la donnée `siz` sera initialisée directement après le prototype, et avant le bloc d'instructions du constructeur.
4. Déclarer les deux classes comme amies.
5. Modifier la fonction `matcreuse_vec` afin qu'elle prenne en argument une instance de `solution` et renvoie comme résultat une autre instance de cette classe.
6. Ecrire une fonction membre de la classe `matcreuse`, qui prend en argument une instance `b` de `solution`, applique l'algorithme du gradient conjugué à une matrice de cette classe avec `b` comme second membre et renvoie comme résultat une instance de `solution` solution du système linéaire considéré. Tester cette fonction pour la valider.
7. On applique maintenant cette fonction à la résolution de l'équation

$$-u''(x) = f(x) \quad \forall x \in]0, 1[.$$

avec la condition aux limites $u(0) = u(1) = 0$. et $f(x) = 1$. Quelle est la solution exacte de cette équation ?

8. Modifier le programme pour calculer la solution approchée de cette équation.
9. Ecrire une fonction qui calcule l'erreur en norme infinie entre la solution exacte et la solution approchée. Calculer cette erreur pour $N = 10, 50, 100, 200$.
10. Créer un nouveau constructeur de la classe `matcreuse`, qui prend en argument un entier N et un réel dt et crée la matrice de la discrétisation implicite de l'équation de la chaleur en 1D.
11. Ecrire une fonction membre de la classe `solution`, amie de la classe `matcreuse`, qui prend en argument une instance de `solution` représentant la solution approchée de l'équation de la chaleur au temps n , et renvoyant la même instance modifiée pour contenir la solution approchée au temps $n + 1$.
12. Inclure dans le programme principal une boucle pour calculer plusieurs itérations successives de la résolution implicite de l'équation de la chaleur.
13. Sauvegarder la condition initiale de l'équation de la chaleur et le résultat final (une fois les itérations finies) dans un fichier, qui contient trois colonnes : position du point, valeur initiale et valeur finale.
14. Afficher le résultat avec le logiciel gnuplot.