

PIF1006 - Mathématiques pour informaticiens II

Session Automne 2022

TRAVAIL #2

Indications principaux	
Date de présentation de l'énoncé	4 novembre 2022
Date de remise du travail	9 décembre 2022 à 23h59
Politiquesurremises tardives	-25% par jour de retard
Éléments et format de remise	Rapport et fichiers complet de la solution ou du projet remis sous forme électronique sur le portail de cours dans un seul fichier compressé (.zip)
Pondération	10%
Nb d'étudiants par équipe	1 à 5, <u>sans exception</u> (une équipe de 2 ou 3 est conseillée)

INTRODUCTION

Dans ce deuxième travail pratique, vous mettrez à épreuve vos compréhensions des opérations sur les matrices et les systèmes de n équations et n inconnues

Vous devrez vous appliquer à implémenter le nécessaire pour pouvoir résoudre des systèmes d'équations en appliquant les méthodes de Cramer, Gauss et de la matrice inverse, ce qui nécessitera aussi d'implémenter des méthodes qui permettent de calculer les déterminants, les transposées, les comatrices, les matrices inverse et plus encore.

L'objectif est d'aiguiller votre capacité à pouvoir traduire des opérations mathématiques d'une certaine complexité en un algorithme.

DESCRIPTION DÉTAILLÉE DU TRAVAIL À ACCOMPLIR

Vous devez créer une application console ou avec une interface graphique (à votre guise) dans laquelle un utilisateur se voit offrir différentes options qui lui permettront de construire des systèmes et d'en tester la résolution avec les trois méthodes vues en classe.

Vous devrez :

- Créer des fichiers textes/JSON/XML ou un format de votre choix dans un format désérialisable/interprétable facilement d'un point de vue algorithmique;
- Permettre l'affichage du système et de ses matrices;
- Permettre de résoudre avec les trois méthodes et démontrer que chaque méthode mène au même résultat.

Une solution de base (avec des classes) vous est fournie et doit être exploitée/ajustée, ce qui constituera votre solution, sachant que :

- Toutes les instructions plus précises s'y trouvent;
- Vous pouvez utiliser Java et copier/coller le code de ces classes et adapter;
- Vous devriez inscrire le nom de vos coéquipiers en entête du fichier Program.cs.

Tous les détails sur quoi et comment implémenter s'y trouvent.

AUTRES EXIGENCES/INDICATIONS SUPPLÉMENTAIRES :

- Assurez-vous que vous avez des commentaires pertinents dans votre code.
- Assurez-vous de tester votre application avant de la rendre en vous assurant qu'elle peut rouler sans planter. S'il est impossible pour le correcteur de tester des options dans l'application, il n'aura pas comme mandat de débbugger et recompiler votre application et les points liés au bon fonctionnement de tout ce qui en découle seront perdus.
- Assurez-vous de remettre tous les fichiers de projet/solution, les fichiers de code ET les fichiers compilés de telle sorte que le correcteur puisse ouvrir votre projet dans l'environnement de développement intégré sans devoir créer de projet à partir de vos fichiers.
- À moins d'une entente particulière, vous DEVEZ utiliser C# .Net ou Java.

RAPPORT À REMETTRE

En plus de votre solution complète, votre équipe doit remettre un rapport contenant minimalement :

- 1 page de présentation;
- 1 page détaillant le rôle joué par chacun des membres de l'équipe;
- 1 section sur les problèmes et difficultés rencontrées s'il y a lieu;
- 1 section dans laquelle vous présentez un **guide utilisateur**, **imprimés-écran à l'appui**, qui non seulement détaille comment fonctionne votre application sous la forme d'une preuve de tests en décrivant au minimum pour chaque test :
 1. Le test réalisé;
 2. Les étapes réalisées incl. les entrées utilisées
 3. Les résultats tels qu'attendus (si vous n'avez pas réussi, vos hypothèses et la liste de ce que vous avez tentés de faire)

Les tests demandés sont au min. les suivants :

1. Deux exemples de fichiers (leur contenu) vs ce qui apparaît en console une fois le fichier chargé (en affichage de système (équations) et de chacune des matrices du système);
2. Un exemple de fichiers mal formaté et le résultat au retour;
3. Un exemple de système et le résultat retourné pour chacune des méthodes (preuve que c'est le même résultat)
4. Un exemple de résolution par matrice inverse et avec Cramer avec une matrice A qui a un déterminant nul.

GRILLE D'ÉVALUATION

Voici les critères d'évaluation et la pondération associée à chacun d'entre eux (pour chacun des critères sont évalués la fonctionnalité (est-ce que cela fonctionne?) et les structures/algorithmes/formes du code (est-ce bien fait et lisible?):

<i>Sujet d'évaluation</i>	<i>Pts (/10)</i>
Rapport complet/guide utilisateur/preuves de tests	1
Menu / interactions utilisateurs/trices et affichage en console	2
Entité Matrice	4
Entité Système	3