



Institut Supérieur d'Informatique (ISI)

Année: 2025-2026

Département: Génie Informatique

Filière: GL (Groupe 3)

Module: RECHERCHE OPÉRATIONNELLE

Niveau: LICENCE 2 / S3

Enseignant: Dr. Cheikh GUEYE

DEVOIR DE PROGRAMMATION LINÉAIRE - 2 HEURES

Exercice 1 . Modélisation avancée d'un projet logiciel (6 points)

Une entreprise de génie logiciel développe trois types de modules :

- **Module A:** Application Web.
- **Module B:** Application Mobile.
- **Module C:** Composant embarqué.

Chaque module consomme des ressources limitées :

| Ressources | Module A | Module B | Module C | Capacité disponible |
|--------------------|----------|----------|----------|---------------------|
| Heures développeur | 5 | 4 | 6 | 480 |
| Heures test | 3 | 2 | 4 | 300 |
| Heures serveur | 4 | 3 | 5 | 360 |

Les bénéfices (en milliers d'unités monétaires) sont: A : 35, B : 30 et C : 50.

Contraintes supplémentaires:

- Le nombre de modules mobiles doit représenter au moins 30 % du nombre total de modules développés.
- Le nombre de modules embarqués ne peut pas dépasser la moitié du nombre de modules Web.

TRAVAIL DEMANDÉ:

1. Définir les variables de décision.
2. Construire la fonction objectif.
3. Écrire toutes les contraintes (ressources et contraintes logicielles).
4. Mettre le modèle sous forme canonique de programmation linéaire.
5. Expliquer brièvement le sens de chaque contrainte.

Exercice 2 . Méthode graphique approfondie (7 points)

Un chef de projet doit répartir le temps entre deux fonctionnalités logicielles critiques:

- x_1 : nombre de fonctionnalités backend.
- x_2 : nombre de fonctionnalités frontend.

$$(P) \begin{cases} \text{Max } Z = 70x_1 + 50x_2 \\ 4x_1 + 3x_2 \leq 240 & (\text{temps de développement}) \\ 2x_1 + x_2 \leq 100 & (\text{temps de test}) \\ x_1 - x_2 \leq 10 & (\text{équilibre logiciel}) \\ x_1, x_2 \geq 0 \end{cases}$$

TRAVAIL DEMANDÉ:

1. Représenter graphiquement l'ensemble des contraintes.
2. Déterminer analytiquement les sommets de la région réalisable.
3. Justifier que le problème admet une solution optimale.
4. Déterminer graphiquement la solution optimale.
5. Donner la valeur maximale de Z et interpréter le résultat en termes de planification logicielle.

Exercice 3 . Méthode du simplexe complète (7 points)

Une équipe de génie logiciel doit planifier trois types de tâches:

- x_1 : développement de nouvelles fonctionnalités.
- x_2 : correction de bugs.
- x_3 : optimisation du code.

$$(P) \begin{cases} \text{Max } Z = 6x_1 + 4x_2 + 5x_3 \\ 3x_1 + 2x_2 + 2x_3 \leq 180 & (\text{temps développeur}) \\ 2x_1 + x_2 + 2x_3 \leq 140 & (\text{temps test}) \\ x_1 + x_2 + x_3 \leq 100 & (\text{temps machine}) \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

TRAVAIL DEMANDÉ:

1. Mettre le problème sous forme matricielle.
2. Introduire les variables d'écart et écrire la forme standard.
3. Construire le tableau initial du simplexe.
4. Appliquer toutes les itérations du simplexe jusqu'à l'optimalité.
5. Identifier les variables de base à l'optimum.
6. Donner la solution optimale et la valeur maximale de Z .
7. Interpréter le résultat du point de vue de la gestion d'un projet logiciel.

BONNE CHANCE!