

## Département Génie informatique

### Licence Analytique des Données

Module “ Structures des données avancées et Théorie des graphes ”

### Projet de fin de module

#### 1. Liste des projets

N°	Projet	Domaine d'application	Objectif	Points clés à développer	Contenu du rapport	Notions de théorie des graphes
1	<b>Analyse de centralité dans un réseau social</b>	Réseaux sociaux	Identifier les utilisateurs les plus influents dans un petit réseau (20–30 utilisateurs)	Construire le graphe, calculer centralités (degré, proximité, intermédiarité, PageRank), interpréter résultats	Introduction, description du graphe méthodes de centralité, résultats, analyse et conclusion	Graphe simple, centralité, degré, proximité, intermédiarité, PageRank
2	<b>Détection de communautés dans un graphe social</b>	Réseaux sociaux	Identifier des groupes cohérents d'utilisateurs	Construire le graphe, appliquer Louvain ou Girvan–Newman, analyser cohésion interne et relations entre communautés	Introduction, présentation du graphe, algorithme choisi, résultats, analyse, conclusion	Graphe simple, sous-graphes, détection de communautés, modularité

3	<b>Système de recommandation d'amis</b>	Réseaux sociaux	Proposer des amis selon similarité (nombre de voisins communs, Jaccard)	Définir métriques, expliquer calcul, tester sur un petit graphe (20–30 nœuds)	Introduction, description du graphe, métriques, résultats et recommandations, analyse critique, conclusion	Graphe simple, voisinage, similarité, métriques de proximité
4	<b>Mini Web et calcul PageRank</b>	Web / Moteurs de recherche	Identifier les pages les plus importantes dans un mini-Web de 10–15 pages	Construire graphe des pages, calculer PageRank, interpréter résultats	Introduction, modélisation, méthode PageRank, résultats et visualisation, analyse, conclusion	Graphe orienté, PageRank, degrés entrants/sortants
5	<b>Analyse d'un graphe de site Web</b>	Web / Moteurs de recherche	Étudier structure interne d'un site : pages centrales, isolées, cycles	Construire graphe, analyser structure, proposer recommandations	Introduction, présentation du site et graphe, méthodes d'analyse, résultats et visualisation, recommandations, conclusion	Graphe orienté, connectivité, cycles, nœuds isolés
6	<b>Recommandation de films via graphe biparti</b>	Systèmes de recommandation	Proposer des films à un utilisateur	Construire matrice utilisateurs–films, calculer similarité (cosinus, Jaccard), justifier recommandations	Introduction, description données, construction matrice, méthode similarité, résultats, analyse critique, conclusion	Graphe biparti, voisinage, similarité, matrices d'adjacence
7	<b>Recommandation de produits façon Amazon</b>	Systèmes de recommandation	Identifier produits pertinents pour un utilisateur à partir d'un historique fictif	Construire graphe utilisateurs–produits, détecter patterns, vérifier cohérence	Introduction, description graphe, méthode similarité, résultats et visualisation, analyse critique, conclusion	Graphe biparti, voisinage, similarité, parcours de voisinage
8	<b>Algorithme de Dijkstra</b>	Transport / Logistique	Trouver le chemin le plus court entre	Construire graphe pondéré, calculer	Introduction, modélisation du	Graphe pondéré, chemin le plus

			deux points dans un réseau de routes	chemin optimal, expliquer la logique	réseau, algorithme utilisé, résultats, analyse et conclusion	court, distances, algorithme de Dijkstra
9	<b>Optimisation du réseau de bus</b>	Transport / Logistique	Réduire les temps de parcours	Pondérer lignes de bus selon distance/temps, proposer itinéraires alternatifs, comparer avec trajets existants	Introduction, description réseau, méthode optimisation, résultats et comparaisons, analyse améliorations, conclusion	Graphe pondéré, chemin optimal, distances/temps, optimisation
10	<b>Réseau ferroviaire</b>	Transport / Logistique	Minimiser temps d'attente et optimiser correspondances	Identifier gares et connexions, plan optimal, proposer améliorations	Introduction, modélisation, méthode optimisation, résultats, analyse et propositions, conclusion	Graphe pondéré, chemins optimaux, sous-graphes, correspondances
11	<b>Détection d'activités suspectes dans un réseau</b>	Cybersécurité	Identifier chemins anormaux ou cycles inhabituels	Étudier structure du graphe, détecter anomalies, interpréter comportements	Introduction, description réseau, méthodes de détection, résultats et visualisation, analyse critique, conclusion	Graphe orienté/non orienté, cycles, motifs anormaux, centralité
12	<b>Surveillance des connexions réseau</b>	Cybersécurité	Identifier nœuds suspects pouvant indiquer attaque	Identifier nœuds anormalement connectés, analyser risques	Introduction, description réseau, méthodes de détection, résultats et visualisation, analyse risques, conclusion	Graphe orienté/non orienté, degrés des nœuds, motifs anormaux
13	<b>Réseau de gènes</b>	Bio-informatique	Identifier sous-graphes fonctionnels	Construire graphe interactions gènes, détecter sous-graphes,	Introduction, description données, modélisation réseau, résultats et	Graphe simple, sous-graphes, motifs, centralité

				interpréter biologiquement	visualisation, analyse biologique, conclusion	
14	<b>Réseau de protéines (PPI)</b>	Bio-informatique	Identifier protéines clés via mesures de centralité	Calculer centralités, visualiser graphe, interpréter biologiquement	Introduction, description réseau, méthodes de centralité, résultats et visualisation, analyse biologique, conclusion	Graphe simple, centralité, degré, intermédiarité
15	<b>Détection de sous-groupes dans un réseau d'amis</b>	Réseaux sociaux	Identifier des groupes d'amis cohérents dans un petit réseau (20–30 utilisateurs)	Construire le graphe des relations, appliquer un algorithme simple (ex. Louvain), visualiser les groupes, analyser relations internes/externe	Introduction, description du graphe, algorithme utilisé, résultats et visualisation, analyse des groupes, conclusion	Graphe simple, sous-graphes, détection de communautés
16	<b>Visualisation des relations dans un mini réseau social</b>	Réseaux sociaux	Créer une visualisation 2D/3D d'un petit réseau pour détecter des utilisateurs centraux ou groupes	Construire graphe (20–30 nœuds), appliquer techniques simples de layout ou réduction de dimension (PCA/t-SNE), analyser patterns	Introduction, description du graphe, méthode de visualisation, résultats et visualisation, analyse, conclusion	Graphe simple, positions relatives, layout, représentation visuelle
17	<b>Classification simple de profils selon activité</b>	Réseaux sociaux / Systèmes de recommandation	Classifier les utilisateurs selon des caractéristiques simples (nombre d'amis, activité, centralité)	Extraire features simples pour chaque nœud, appliquer un algorithme de classification basique (k-NN, arbre de décision), analyser performance et interpréter résultats	Introduction, description des profils et features, méthode de classification, résultats, visualisation, analyse, conclusion	Graphe simple, features locales (degré, centralité), classification de nœuds

## Éléments à rendre

1. **Rapport écrit** : Word ou PDF, bien structuré (introduction, description du graphe, méthodes, résultats, analyse, conclusion).
2. **Présentation PPT** : synthèse des résultats, visualisations, points clés du projet.
3. **Code source**
  - Le code doit être **correctement commenté et fonctionnel**.
  - Il doit inclure **des explications sur l'implémentation et les résultats**.
  - Les étudiants **ont le choix du langage de programmation**, par exemple : **C, Python, Java, ou autre langage adapté**.
  - Les instructions pour **exécuter le code et reproduire les résultats** doivent être fournies (README ou section dans le rapport).

## Dates importantes

- **Date de soumission des rapports 18 janvier 2026**
- **Date de présentation 20 -21-22 Janvier 2026**