

Projet 1 : pluridisciplinaire pour les 2^{ème} année ingénieur

Titre du sujet proposé : Conception et réalisation d'une application informatique (SI) pour la génération automatique des emplois du temps de la faculté informatique.

Résumé :

L'objectif principal de ce travail est de faciliter la tâche de la faculté informatique en matière d'emploi du temps en utilisant l'outil informatique, à travers la réalisation d'une application permettant :

- D'établir des emplois du temps pour la faculté informatique sans conflit au niveau des séances ;
- De consulter et suivre rapidement les emplois du temps à tout moment ;
- De diminuer considérablement le temps d'attente (délivrance automatique) ;
- De garantir une meilleure coordination du travail.

Pour ce faire, les étudiants doivent suivre les étapes suivantes :

- Procéder à une étude générale du système pédagogique de la faculté informatique ;
- La conception de l'application, soit en utilisant la méthode MERISE ou/et le langage de modélisation UML ;
- Le développement de l'application, tout en décrivant les outils de programmation utilisés, les interfaces graphiques développées ainsi que les différentes fonctionnalités de l'application.

L'application informatique considéré doit contenir plusieurs interfaces permettant l'accès facile, rapide et simple aux informations, notamment :

- 1- La saisie des informations d'une nouvelle séance, des informations des nouveaux enseignants, groupes, modules, salles et matières.
- 2- La modification des informations qui concernent les enseignants, les groupes, les modules, les salles.
- 3- L'affichage de l'emploi du temps pour les niveaux L1, L2, L3, M1, M2 et Ingénieur.
- 4- La suppression : suppression d'une séance, d'un enseignant, d'un groupe, d'un module, d'une salle.
- 5- La mise à jour de la base de données.

L'application informatique sera implémentée en combinant des langages de programmation pour la création des interfaces graphique et un SGBD pour la gestion de la base de données.

Projet : Architecture des Ordinateurs – Programmation Assembleur

Sujet : Jeu de Dames

On s'intéresse dans ce projet au jeu de dames qui se joue sur une grille (matrice) de taille 10 x 10. Les pièces dans ce jeu sont le pion et la dame, qui peuvent être aussi bien blancs que noirs. Initialement, toutes les pièces sont des pions. Un pion se déplace dans un seul sens, et devient une dame lorsqu'il atteint la dernière ligne. La figure suivante présente la numérotation Manoury du damier, la position des pièces en début d'une partie, ainsi qu'une partie en cours.

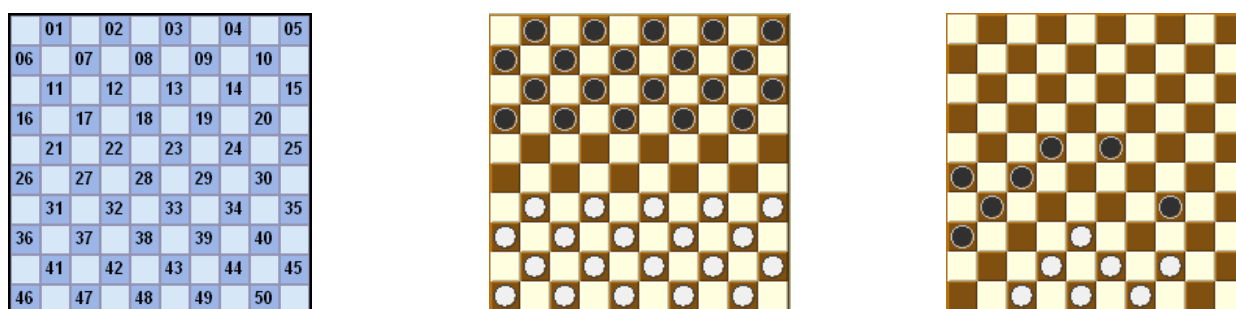


Figure 1 : Représentation matricielle du jeu de Dames, Disposition initiale, et une partie en cours.

Une partie du jeu de dames peut donc être stockée dans une matrice de taille 10 x 10, où chaque case contient un pion blanc, pion noir, dame blanche, dame noire ou elle peut être libre (5 valeurs possibles). Néanmoins, les pièces se déplacent uniquement sur les cases noires (cases blanches ne sont jamais utilisées). Donc, pour économiser de l'espace mémoire, on propose de passer de la numérotation Manoury (matricielle) à une numérotation vectorielle dans laquelle on stocke juste les cases de 1 à 50.

Partie Algorithmique du projet:

- Écrivez une fonction qui retourne la ligne $1 \leq i \leq 10$ d'un numéro **N** entre 1 et 50.
Exemple : Pour **N=9**, alors **i=2**. Pour **N=0** ou **N=52**, erreur car en dehors de l'intervalle [1, 50].
- Écrivez une fonction qui retourne la colonne $1 \leq j \leq 10$ d'un numéro **N** entre 1 et 50.
Exemple : Pour **N=9**, alors **J=7**.
- Écrivez maintenant une fonction qui retourne le numéro **N** correspondant à la ligne **i** et la colonne **j**. Exemple : pour **i=j=3**, Erreur car c'est une case blanche. Si **i=3**, **j=4**, alors **N=12**.
- Traduisez le code des trois fonctions précédentes en Assembleur 8086 pour les réutiliser plus tard dans votre projet.

Partie Assembleur 8086 :

On s'intéresse maintenant à l'implémentation en Assembleur 8086 de seulement quelques fonctionnalités du jeu de Dames. La structure utilisée pour stocker le jeu doit être un tableau.

- e. Détaillez la structure de données utilisée pour stocker la partie de jeu de dames, et donnez le code qui alloue une telle structure en mémoire.
- f. Ajoutez une fonction *Initialiser-Damier* à votre programme pour initialiser les cases du damier telle que représenté dans la figure 1.
- g. Écrivez une fonction qui prend en arguments les indices d'une case **i** et **j** et affiche la couleur de cette case (c-à-d, blanche ou noire).
- h. Écrivez une fonction qui prend en arguments les indices d'une case **i** et **j** et affiche l'état de la case (c-à-d, le type du pion l'occupant et sa couleur, ou bien *libre*).
- i. Ajoutez une fonction qui affiche sous forme d'une matrice le damier complet case par case tout en tenant compte de la couleur et du contenu des cases.

Déplacement d'un pion : On s'intéresse au déplacement des pions (et non des dames). Le sens de déplacement d'un pion est toujours vers le camps adverse (pas de retour arrière), et le déplacement se fait toujours vers une case libre, et peut être direct ou indirect. Supposons, par exemple, que la case numéro 13 est occupée par un pion noir. Les déplacements possibles d'un tel pion sont :

- **Direct :** Vers les cases 18 et 19 si ces dernières sont libres, et le déplacement s'arrête.
 - **Indirect :** Si la case 18 (resp. 19) est occupée par un pion ou une dame adverse (c-à-d blancs), un déplacement est possible vers la case 22 (resp. 25) si cette case 22 (resp. 25) est libre. Le pion adverse traversé pendant le déplacement doit être retiré du jeu.
 - **Remarque :** Dans le cas général, un pion peut traverser 3 pions adverses, mais dans notre projet on se limite à traverser 1 seul pion adverse.
- j. Ajoutez une fonction qui prend en arguments les indices d'une case de départ (**i, j**) et les indices d'une case d'arrivée (**x, y**) et affiche si le déplacement du pion se trouvant à (**i, j**) vers (**x, y**) est possible ou non. Dans cette fonction, on suppose que la case (**i, j**) est soit occupée par un pion ou elle est libre (pas de dame dans la case). Dans le cas où la case est libre, la fonction doit indiquer à l'utilisateur qu'aucun déplacement n'est effectué.
 - k. Ajoutez une fonction qui prend en arguments les indices d'une case de départ (**i, j**) et les indices d'une case d'arrivée (**x, y**) et effectue un déplacement du pion se trouvant à (**i, j**) vers (**x, y**) si possible.

Ajouter une fonction permettant de proposer un menu de choix à l'utilisateur où il pourra choisir une des fonctions précédentes. L'utilisateur doit pouvoir quitter le programme à tout moment.

La liste des fonctions demandées n'est pas exhaustive. Vous pouvez l'étendre pour améliorer votre programme. Vous pouvez également implémenter les déplacements longs (en traversant 2/3 pions adverses) par exemple. Toute fonctionnalité supplémentaire sera récompensée.

Organisation : Le projet est à réaliser par groupe de 4 à 5 étudiants au maximum.

Votre code source implémenter sous l'émulateur 8086 (bien évidemment correctement commenté et documenté), ainsi qu'un rapport d'au maximum 10 pages, contenant principalement vos choix d'implémentation et la description de vos fonctions, doivent être envoyés par mail à l'adresse projetsinfoing2@gmail.com sous la forme d'une archive ayant comme nom **ProjetArchitecture-Numéro.zip** où **Numéro** est le numéro du binôme (qui vous communiqué ultérieurement).

L'objet du mail doit être **Projet Pluridisciplinaire : Architecture**.

Projet 3 : Algorithmique et Complexité

La tour de Hanoï

Dans ce mini projet vous avez la tâche d'implémenter, simuler et analyser une méthode de résolution exacte d'un problème NP-complet.

L'objet de ce projet est l'étude expérimentale de l'algorithme de résolution du problème des "tours de Hanoï", qui est un problème classique en informatique.

Partie I :

1. Implémenter l'algorithme de résolution de la tour de hanoï en version récursif et itératif en langage C.

Partie II :

Pour cette partie du projet, votre rapport doit traiter les points suivants :

- i. **Etude théorique du problème :**
 - a. Historique et présentation du problème.
 - b. Définition formelle du problème.
 - c. Présenter la modélisation de la solution (Structure de données de la solution).
 - d. Présenter l'algorithme de résolution avec calcul détaillé de sa complexité théorique.
 - e. Présentation l'algorithme de vérification avec pseudo-code et calcul détaillé de sa complexité théorique et spatiale.
 - f. Présenter une instance du problème avec sa solution (un exemple avec déroulement).

ii. Etude Expérimentale : (Variation de la taille du problème)

Nombre de disque (n)	5	10	15	20	25	...	90	95	100
Temps d'exécution									
Nombre de déplacements effectués									

- a. Simuler la complexité temporelle et spatiale **théorique** de l'algorithme de résolution (itératif et récursif).
 - b. Simuler la complexité temporelle et spatiale **théorique** de l'algorithme de vérification.
 - c. A quoi correspond le meilleur, moyen et pire cas pour chaque algorithme ?
Justifiez
 - d. Analyse des résultats.
- iii. Conclusion
 - iv. Références
 - v. Annexe : code source

Directives :

- Le travail doit se faire en quadrinôme.
- Inclure la **distribution des tâches** entre les membres du groupe dans le rapport après la conclusion.
- Un rapport version numérique (pdf) doit être envoyé vers l'adresse suivante :
hw.moulai@gmail.com avec comme **objet de l'email** : **L2-ING-Hanoï-NomsBinome**
- Le deadline est fixé pour le **/ /2024**.
- **Tout ajout ou initiative peut être récompensé !**

RÈGLE DU JEU

Le départ

On dispose de 3 piquets fixés sur un socle, et d'un nombre n de disques de diamètres différents. Les disques sont empilés sur un piquet, en commençant du plus large au plus petit. Le nombre de disques peut varier. Plus il y a de disques au départ, plus le jeu est difficile.

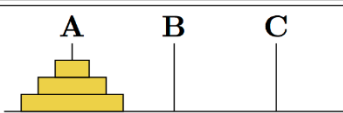
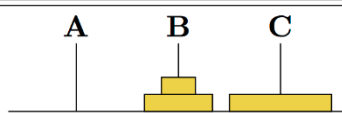
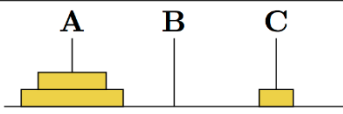
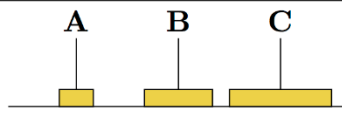
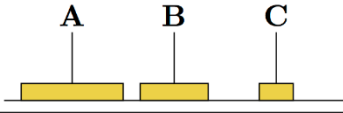
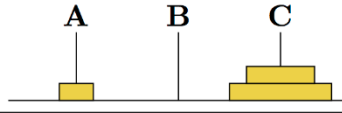
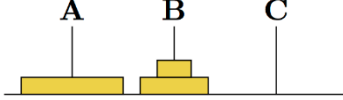
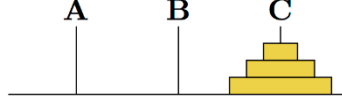
Le but

Déplacer les disques d'une tour de 'départ' à une tour 'd'arrivée' en passant par une tour 'intermédiaire', et ceci en un minimum de coups.

Comment

2 règles simples :

- on ne déplace qu'un seul disque à la fois, et le disque déplacé doit l'être sur l'un des deux autres piquets au choix ; c'est ce que l'on appelle un déplacement.
- le disque déplacé ne doit jamais être placé au-dessus d'un disque plus petit que lui.

Mouvement	Position	Mouvement	Position
Position initiale		4 : A vers C	
1 : A vers C		5 : B vers A	
2 : A vers B		6 : B vers C	
3 : C vers B		7 : A vers C	

RÉSOLUTION

La solution générale est donnée par l'algorithme suivant.

Algorithme récursif

La solution de base du jeu de la Tour de Hanoï est formulée de manière récursive. Étiquetez les piquets avec A - B - C et numérotez les disques de 1 (le plus petit) à n (le plus grand). L'algorithme est exprimé comme suit :

1. Déplacez les $n-1$ premiers disques de A à B. (Cela laisse le disque n seul sur le piquet A)
2. Déplacez le disque n de A à C
3. Déplacez les $n-1$ disques de B à C

Pour déplacer n disques, il faut effectuer une opération élémentaire (déplacement d'un seul disque) et une opération complexe, c'est-à-dire le mouvement de $n-1$ disques. Cependant, cette opération se résout également de la même manière, en demandant comme opération complexe le déplacement de $n-2$ disques. En itérant ce raisonnement on réduit le processus complexe à un processus élémentaire, c'est-à-dire le déplacement de $n-(n-1)=1$ disque.

Proposition de thème de projet pluridisciplinaire

Sujet 4 : Conception et implémentation d'une plateforme de E-learning visuel.

Il s'agit de concevoir et implémenter une plateforme ayant comme but de présenter les matières enseignées d'une façon visuelle et dynamique. Les étudiants vont choisir une matière telle que les bases de données. En étudiant cette matière, ils vont proposer des mécanismes de concrétisation de ses concepts, notamment à travers des animations et des visualisations. De plus, ils peuvent proposer des quiz pour l'évaluation des apprenants dans la matière. Egalement, on peut implémenter des fonctionnalités standards dans les plateformes E-learning, telles que l'inscription et le suivi d'état d'avancement.

Technologies proposées

Modélisation : UML

Base de données : PostgreSQL / MongoDB

Front-End : JavaScript (React)

Back-End : JavaScript (ExpressJS)

Collaboration : Git (Github)

Indice

Le site web ci-dessous offre un travail similaire, sur la partie de visualisation, à celui demander dans ce projet.

<https://visualgo.net>

Sujet 5 : Conception et implémentation d'un outil d'estimation des prix des véhicules.

C'est un sujet qui entre dans le cadre de la science des données. Le projet comporte plusieurs phases :

1. La collecte des données : principalement à travers le développement d'un outil de web scraping pour récupérer des données sur des plateformes ayant des données sur les prix des véhicules
2. Le prétraitement de données : l'état des données collectées est généralement assez médiocre et nécessite un prétraitement avant qu'elles soient utilisées pour entraîner des modèles de machine learning.
3. L'entraînement des modèles : en utilisant des bibliothèques avec des modèles de machine learning implémentés, on cherche à choisir le meilleur modèle pour l'estimation.
4. Le développement d'une interface graphique : une interface web avec principalement un formulaire pour saisir les données d'un véhicule et fournir une estimation de son prix.

Technologies proposées

Modélisation : UML

Data sets : fichiers CSV

Front-End : JavaScript

Back-End : Python

Collaboration : Git (Github)

Projet Pluridisciplinaire – Ingénieur Informatique 2ème Année

Dr. LAHRECHE Abdelmadjid

Sujet : Conception et Réalisation d'une Application Web de Médias Sociaux

L'essor des réseaux sociaux a transformé la manière dont les gens interagissent, communiquent et partagent des informations en ligne. Ce projet vise à concevoir et à développer une application de médias sociaux fonctionnelle et conviviale, permettant aux utilisateurs de créer des profils, de partager du contenu, d'interagir avec d'autres utilisateurs et de découvrir du contenu personnalisé.

Fonctionnalités clés de l'application

- Création de profil utilisateur avec des informations telles que le nom, l'avatar, la biographie, etc.
- Publication de contenu : texte, images, vidéos, liens, etc.
- Possibilité de commenter, aimer et partager des publications.
- Fonction de recherche pour trouver d'autres utilisateurs et du contenu.
- Système de notifications pour informer les utilisateurs des activités telles que les mentions, les commentaires sur leurs publications, etc.
- Système de suivi des autres utilisateurs pour voir leur contenu dans un fil d'actualité personnalisé.
- Gestion sécurisée des données utilisateur, y compris le stockage sécurisé des mots de passe et des informations sensibles.
- Contrôles de confidentialité permettant aux utilisateurs de gérer qui peut voir leur contenu et leurs informations personnelles.

Plan de travail

1. Définition des exigences du projet

- Identifier les fonctionnalités de base et avancées de l'application.
- Définir les besoins en termes de performances, de sécurité et de convivialité.

2. Conception de la base de données

- Définir le schéma de la base de données en identifiant les entités principales telles que les utilisateurs, les publications, les commentaires, etc.
- Concevoir les relations entre les entités et les contraintes d'intégrité référentielle.
- Choisir le type de base de données à utiliser (relationnelle, orienté objet, etc.) en fonction des besoins du projet.

3. Développement de l'application

- Créer une architecture logicielle pour l'application, en incluant les composants Frontend et Backend.
- Mettre en place le système d'authentification et de gestion des sessions utilisateur.
- Implémenter les fonctionnalités de base telles que la création de profil, la publication de contenu, la gestion des commentaires, etc.
- Assurer la sécurité de l'application en appliquant les meilleures pratiques de développement sécurisé.

The University of Science and Technology Houari Boumediene

Computer science faculty

Multidisciplinary project 2nd year engineering

By:Dr.Haouari Ahmed Taha

Introduction to Textual Gaming

Textual gaming, often referred to as interactive fiction or text-based gaming, represents a captivating form of interactive entertainment where the narrative unfolds primarily through written text. Unlike traditional graphical games that rely heavily on visual and auditory stimuli, textual games leverage the power of language and imagination to immerse players in richly detailed worlds, compelling storylines, and complex decision-making scenarios. In a textual game, players navigate through the story by reading descriptive passages, making choices that impact the progression of the narrative, and interacting with characters and objects through text-based commands.



These games offer a unique blend of storytelling, exploration, and puzzle-solving, challenging players to use their wit and creativity to overcome obstacles and uncover hidden secrets. Textual gaming traces its roots back to the early days of computer programming, with classic examples such as "Adventure" (also known as "Colossal Cave Adventure") pioneering the genre in the 1970s. Since then, textual gaming has evolved significantly, with modern titles incorporating sophisticated narrative structures, dynamic character interactions, and branching storylines that respond to player choices in real-time.

Despite the rise of graphical gaming technologies, textual gaming continues to thrive, attracting a dedicated community of enthusiasts and indie developers who appreciate its emphasis on narrative depth, player agency, and intellectual engagement. Textual games offer a platform for experimentation, exploration, and storytelling freedom, allowing creators to craft immersive experiences that transcend the limitations of visual representation.

Now, with an understanding of the essence and appeal of textual gaming, let's delve into the details of our comprehensive text-based game project, "Textual Adventures," where students will have the opportunity to explore this captivating genre while honing their programming skills and unleashing their creativity.



Project Details: "Textual Adventures" is a comprehensive computer science project that invites second-year students to immerse themselves in the world of textual gaming, where they will create their own scenarios within the framework of interactive fiction. In addition to mastering core programming concepts and techniques, students will have the option to implement a graphical interface to enhance the player experience.

Key components and implementation strategies include:

1. **Scenario Creation:** Students will craft compelling narratives, diverse characters, and intricate worlds, leveraging their creativity and storytelling skills to engage players in immersive textual adventures.
2. **Optional Graphic Interface:** While the focus is on textual interaction, students may choose to incorporate a graphical interface using suitable libraries or frameworks, enriching the game with visual elements that complement the textual descriptions.
3. **Object-Oriented Design:** Students will design the core architecture of their game using object-oriented programming principles, facilitating modularity, extensibility, and scenario customization.

4. **Data Structures and Algorithms:** Essential data structures and algorithms will be implemented to manage game state, handle player interactions, and facilitate scenario-specific logic, ensuring smooth gameplay experiences.
5. **User Interface Development:** Students will design user-friendly interfaces that provide clear feedback, intuitive navigation, and immersive storytelling, whether through textual interaction or graphical enhancements.

Resources to Aid Students:

To support students in their "Textual Adventures" project, a variety of resources can be made available:

1. **Text-Based Game Engines:** Provide access to popular text-based game engines and frameworks such as Inform, Twine, or ChoiceScript, along with tutorials and documentation to facilitate scenario creation and game development.
2. **Programming Libraries:** Offer libraries and APIs for implementing optional graphical interfaces, such as Pygame for Python or LibGDX for Java, along with tutorials and examples to help students integrate visual elements into their games.
3. **Online Communities and Forums:** Encourage students to participate in online communities and forums dedicated to textual gaming and game development, where they can seek advice, share ideas, and collaborate with peers and experienced developers.

By providing access to these resources and fostering a supportive learning environment, students will have the guidance and tools they need to successfully complete their "Textual Adventures" project and unleash their creativity in the realm of textual gaming.

Good Luck

Projet 8 : Attaques sur les mots de passe

Modules : Algo, Complexité, SE, Fichiers,

Contexte : test de la robustesse des mots de passe

L'objectif de ce projet est de développer une application permettant de découvrir les mots de passe utilisés lors de l'accès au système ou bien pour l'ouverture des applications et les fichiers protégés. Il s'agit d'appliquer des attaques sur des mots de passe cryptés.

Objectifs :

- Étudier l'état de l'art sur les méthodes de protection des mots de passe (hachage) ainsi que les techniques d'attaques utilisées pour les découvrir.
- Implémenter et tester deux attaques sur les mots de passe cryptés.
- Valider les techniques implémentées sur des jeux de données réels.
- Conclure par des propositions pour renforcer la robustesse des mots de passes.

NB. Des outils open source peuvent être considérés comme exemple typique dans ce projet : voir la distribution *kali linux*.

Langage(s) et technologie(s) :

- C, Python, VMware

Projet pluridisciplinaire pour les 2ème année ingénieur

Titre du sujet proposé : Implémentation d'une application de gestion d'une salle de cinéma.

Travail à faire

L'objectif du projet est de concevoir et implémenter en Java un système de gestion d'une salle de cinéma. Le système doit gérer les salles de diffusion, les films diffusés ainsi que les billets réservés. Une salle de cinéma est caractérisée par son numéro et sa capacité d'accueil. Un film est caractérisé par son nom, sa durée, son genre et sa version (Vo/Vf). Une diffusion comporte le film, la salle, la date ainsi que l'horaire de diffusion. L'horaire comporte une heure de début et une heure de fin. Un billet est caractérisé par son numéro, son prix et la diffusion concernée.

L'application doit comporter une fenêtre d'accueil avec le menu suivant :

1. Gestion des films : qui comporte les fonctionnalités suivantes : Ajouter, Modifier, Supprimer un film et Afficher la liste de tous les films.
2. Gestion des salles de diffusion : qui comporte les fonctionnalités suivantes : Ajouter, Modifier, Supprimer une salle et Afficher la liste de toutes les salles de diffusion.
3. Gestion des diffusions : qui comporte les fonctionnalités suivantes : Ajouter, Modifier, Supprimer une diffusion, Afficher les diffusions d'une date donnée par salle.
4. Gestion des billets : qui comporte les fonctionnalités suivantes : Réserver un billet, Annuler un billet et Afficher le nombre des billets restants pour une diffusion donnée. La réservation de billets dépend du nombre de billets qui est égale à la capacité de la salle concernée par la diffusion. A chaque réservation, un billet est créé et le nombre de billets est décrémenté. Si le billet est annulé, le nombre de billets disponibles s'incrémente. S'il n'y plus de billets disponibles, la réservation ne peut pas se faire.

Projet 10 pluridisciplinaire pour Ingénieur informatique (en POO) 2023/2024

D^r MEKAHLIA Fatma Zohra

Thème : Nouvelles métriques pour un programme orienté objet.

Modules : Algorithmique, Programmation Orienté Objet.

Problématique : la programmation orienté objet est un paradigme de programmation qui repose sur le concept de classes et objets. Les classes sont utilisés pour créer des instances individuelles d'objets. Un objet est une entité qui englobe à la fois son état (qui représente les valeurs des attributs) ainsi que son comportement (qui représente l'implémentation des méthodes de la classe). Dans un programme O.O les objets collaborent les uns avec les autres par l'envoi et la réception des messages afin de réaliser une tâche. Après chaque réalisation d'un produit logiciel, nous devons tester notre produit avant sa livraison au client (test unitaire, test d'intégration ou encore test système) et comme la phase de test souvent plus longue que les phases de spécification, conception et de réalisation réunies, des modèles de prédiction des fautes a fait l'objet de plusieurs travaux actuellement pour assurer la qualité logicielle en utilisant des algorithmes d'apprentissage automatique. Dans ce contexte, Des métriques mesurant un programme orientée objet sont souvent utilisées comme :

- 1) WMC : weighted methods per class
- 2) DIT : Depth In The Tree
- 3) NOC :Number Of Children
- 4) CBO : Coupling Between Objects
- 5) RFC : Reference For Class
- 6) LOC : Lines of Code

7) NSM : Number of Static Methods

Etc,

Travail demandé : Dans ce contexte, on vous demande de:

- Trouver tous les métriques qui existent déjà dans la littérature.
- Etudier les algorithmes et les implémentations de calcul des métriques déjà existents.
- Trouver de nouvelles métriques pour un programme orienté objet.
- Proposer un algorithme et une implémentation en Java qui calcul chaque nouvelle métrique.
- Rédiger le travail dans un rapport de 10 pages.

Bon courage