# Prosit 5 : Concevoir son approche

## Rôles:

Animateur : Raphaël

Secrétaire :

Scribe : Nathan. G

Gestionnaire : Nathan. A

## Mots clés :

* Expressions arithmétiques
* Pile d’exécution
* Opérateurs (Unaire/Binaire)
* Compilation
* STL

## Contexte :

Développer une calculatrice en mode console capable d’interpréter des expressions arithmétiques complexes avec parenthèses et opérateurs, en utilisant des piles pour traiter les symboles.

## Problématique :

Comment concevoir une calculatrice en mode console qui interprète correctement les expressions arithmétiques complexes tout en respectant les principes de la programmation orientée objet ?

## Contraintes :

* POO
* Priorités des opérations
* Pile pour opérandes et opérateurs
* Opération de base (addition, soustraction, multiplication, division)

## Livrables :

* Diagramme UML
* Code
* Documentation du fonctionnement
* Tests Unitaire

## Généralisations :

* Algorithmes d’analyse syntaxique
* Piles
* Structures de données
* Calcule scientifique
* POO

## Plan d’action :

* Concevoir un parseur pour analyser les expressions arithmétiques
* Implémenter deux piles : une pour les opérandes, une pour les opérateurs
* Gérer les priorités des opérations à l'aide de piles
* Utiliser des classes et méthodes pour chaque fonctionnalité (POO)
* Réaliser des tests unitaires pour valider le traitement des expressions

## Plan d’action :

* Définir les mots clés.
* Étudier les ressources sur les piles et le traitement d'expressions.
* Modéliser les classes nécessaires (diagramme UML).
* Implémenter les piles pour gérer opérateurs et opérandes.
* Développer le code pour le traitement des expressions (lecture, calcul).
* Effectuer des tests unitaires sur différents cas d'expressions.
* Documenter le processus et les choix techniques.