



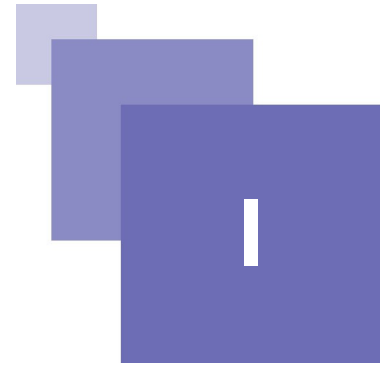
# **Projet : Calculatrice à notation polonaise inverse**

# Table des matières



<b>I - Exercice</b>	<b>5</b>
<b>II - Spécifications</b>	<b>7</b>
<b>III - Notation Polonaise Inverse</b>	<b>11</b>

# Exercice



Vous devez réaliser une calculatrice à notation polonaise inversée (notation postfixe). Vous travaillerez par groupe de deux en collaborant avec l'outil de gestion de versions de votre choix (SVN, CVS, GIT, Mercurial, etc.). Vous veillerez à respecter les principes d'encapsulation de la programmation orientée objet. Vous vous assurerez d'une gestion efficace de la mémoire. Vous ferez particulièrement attention aux designs patterns qu'il conviendra d'utiliser. Vous prendrez garde également à la bonne gestion des erreurs d'exécution.

Vous devrez respecter les spécifications en annexe de ce document. Ces spécifications vous proposent une série de fonctions optionnelles, vous devrez en implémenter **au moins une** de votre choix.

Pour mener à bien le projet, vous utiliserez toutes les bibliothèques standards de votre choix, ainsi que du framework Qt. Vous pourrez également réutiliser des développements de TD ou TP qui vous sembleront utiles.

## Résultats attendus

Le projet prendra fin lors de la dernière séance de TP. Une démonstration de votre exécutable vous sera demandé. Vous devrez livrer :

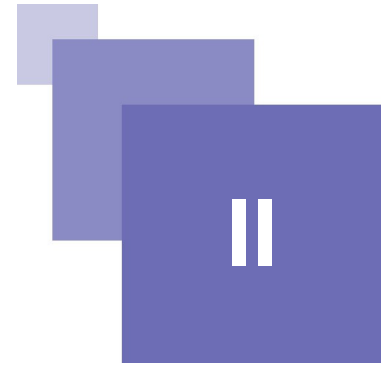
- L'accès au dépôt du système de gestion de versions
- La documentation doxygen complète (version pdf ou un accès à une version web)
- Un document justifiant vos choix de conception, un modèle UML de l'application et les diagrammes de séquence des principaux scénarios (2 à 5 pages)

## Évaluation

La couverture des spécifications donnera une note sur 16 (il sera possible d'obtenir une note jusqu'à 20 en enrichissant votre code par d'autres fonctionnalités). Des points seront enlevés en cas de mauvais choix de conception, manque de clarté dans la documentation ou manque de clarté dans votre code source.

La note du projet s'additionne à une note de participation de TP, sur 4 points (et pour un maximum limité à 20). Le total participe à hauteur de 20% dans la note de l'UV.

# Spécifications



## *Type de données*

---

La calculatrice peut manier différents types de constante :

- Entière : suite de chiffres
- Rationnelle : deux entiers séparé par le caractère '/'
- Réelle : suite de chiffres pouvant accepter une virgule
- Complexe : constituée de deux constante (Entière, Rationnelle ou Réelle) séparées par le caractère '\$'
- Expression : expression entourée du caractère ' à évaluer ultérieurement

## *Gestion des données*

---

- Les constantes sont stockées dans une pile au fur et à mesure de l'exécution
- Les X derniers éléments de la pile sont affichés à l'écran (où X est paramétrable)

## *Évaluation d'une expression*

---

Une expression est évaluée, lorsque l'utilisateur entre une série d'instructions sur la ligne de commandes ou quand une expression (encadrée par le caractère ') de la pile est évaluée avec la commande EVAL (voir ci-après).

- Une expression est composée d'opérateurs et de constantes séparés par des espaces
- Les éléments constituant une expression sont évalués un par un, de la gauche vers la droite
- L'évaluation d'une expression provoque l'empilement du résultat sur la pile
- L'évaluation d'un opérateur binaire provoque le dépilement des deux derniers éléments de la pile sur lesquels sera effectuée l'opération

Si la pile ne contient pas assez de constantes pour évaluer une expression, elle est laissée intacte et la calculatrice indique une erreur. Les éléments de l'expression en cours n'ayant pas encore pu être évalués (dont l'opérateur) sont laissés sur la ligne de commande pour correction.

- Si une constante de type expression est dépilée dans le but d'être utilisée avec un opérateur, les autres éléments constituant l'opération (l'opérateur ainsi que l'éventuel autre argument), sont inclus dans l'expression pour évaluation ultérieure.

Afin d'évaluer une expression, la calculatrice utilisera des modes paramétrables par l'utilisateur :

- Type de constante (entière, rationnelle ou réelle)
- Unité des degrés (radian ou degré)
- Utilisation des complexes (oui ou non)

## Interface

---

- La calculatrice possède un afficheur et un clavier
- L'affichage du clavier est une option paramétrable par l'utilisateur
- La calculatrice peut être utilisée à la souris ou directement au clavier
- Toutes les options de paramétrages sont définies à travers une interface simple et agréable à utiliser.

## Sauvegarde du contexte

---

Au démarrage de l'application, l'état de la pile et les paramétrages de la calculatrices de la dernière exécution sont récupérés.

## Fonctions annuler et rétablir

---

La calculatrice dispose des fonctions « *annuler* » et « *rétablir* ». Ces fonctions peuvent être appelées par un menu ou par les raccourcis Ctrl-Z et Ctrl-Y.

Les types entre parenthèses sont les types disponibles pour l'utilisation de l'opérateur :

- +, -, \*, / (entier, rationnel, réel, complexe)
- POW : puissance (entier, rationnel, réel)
- MOD : modulo (entier)
- SIGN : inversion du signe (entier, rationnel, réel, complexe)

## Opérateurs unaires

---

Les types entre parenthèses sont les types disponibles pour l'utilisation de l'opérateur :

- SIN : sinus (entier, rationnel, réel)
- COS : cosinus (entier, rationnel, réel)
- TAN : tangente (entier, rationnel, réel)
- SINH : sinus hyperbolique (entier, rationnel, réel)
- COSH : cosinus hyperbolique (entier, rationnel, réel)
- TANH : tangente hyperbolique (entier, rationnel, réel)
- LN : logarithme népérien (entier, rationnel, réel)
- LOG : logarithme décimal (entier, rationnel, réel)
- INV : inverse (entier, rationnel, réel)
- SQRT : racine carré (entier, rationnel, réel)
- SQR : fonction carré (entier, rationnel, réel, complexe)
- CUBE : fonction cube (entier, rationnel, réel, complexe)
- ! : factorielle (entier)
- EVAL : évaluation d'une expression (Expression)

## Opérations sur la pile

---

La pile peut être manipulée à l'aide des opérateurs suivants.

Opérateur binaire :

- SWAP : inversion de l'élément numéro x et l'élément numéro y de la pile (où x et y sont les arguments de l'opérateur) (entier)

Opérateurs unaires :

- SUM : somme des x premiers éléments de la pile (où x est l'argument) (entier, rationnel, réel, complexe)

- MEAN : moyenne des x premiers éléments de la pile (où x est l'argument) (entier, rationnel, réel, complexe)

Opérations sans arguments :

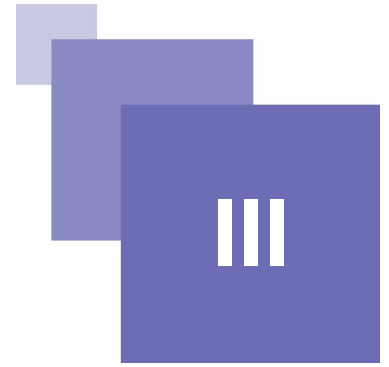
- CLEAR (ou Ctrl - C) : vider la pile
- DUP (ou Entrée sans aucune expression sur la ligne) : duplique le premier élément de la pile
- DROP (ou Retour en arrière sans aucune expression sur la ligne) : supprime le premier élément de la pile

### Options

Pour compléter votre projet, vous choisirez au minimum une des fonctions suivantes :

- Commande utilisateur : vous commencerez par enrichir les expressions par la possibilité d'ajouter une ou deux variables (formatage prédéfini à inclure dans l'expression). Ces expressions s'exécuteront comme un opérateur (unaire ou binaire). Vous ajouterez ensuite un nouvel opérateur binaire prenant une chaîne de caractères et une expression dynamique en argument. L'utilisation de la chaîne appellera l'expression dynamique enregistrée. Vous ferez en sorte que les réglages utilisateurs soient persistants d'une exécution à l'autre de la calculatrice.
- Onglets : vous placerez votre pile dans une collection STL de votre choix. La commande Ctrl-N ouvrira un nouvel onglet avec une pile vide. Cette pile sera ajoutée à la collection. La commande Ctrl - T ouvrira un nouvel onglet en dupliquant la pile courante. La commande Ctrl-W supprimera l'onglet courant.
- Log : vous implémenterez une classe LogMessage, contenant un message et un degré d'importance du message de log. Une seconde classe LogSystem manipulera des LogMessage pour écrire les message de log sur à la console et dans un fichier dédié.

# Notation Polonaise Inverse



La notation polonaise inverse est une méthode de notation mathématique permettant de s'abstenir de l'utilisation de parenthèse.

Lors de la résolution d'une expression exprimée en notation polonaise inversée, un opérateur est remplacé par la valeur de l'opération qu'il représente en utilisant les n termes le précédents (opérateur unaire, un terme, opérateur binaire, deux terme, etc.).

Ainsi :

- $1+1$  s'écrit  $1\ 1\ +$
- $2 \times 2 + 1$  s'écrit  $2\ 2\ \times\ 1\ +$
- $(2 + 3) \times 4$  s'écrit  $2\ 3\ +\ 4\ \times$

Pour plus d'information, vous pouvez consulter la page *wikipédia*<sup>1</sup> dédiée.

1 - [https://fr.wikipedia.org/wiki/Notation\\_polonaise\\_inverse](https://fr.wikipedia.org/wiki/Notation_polonaise_inverse)