

MANUEL D'UTILISATION

1. Introduction	2
1.1. Objectifs et méthodes	2
1.2. Documents de référence	2
2. Guide de lecture	2
3. Concepts de base	2
4. Mise en œuvre	2
4.1. Lancement du logiciel	2
4.2. L'entrée de la fonction principale	2
4.3. Complétion du tableau de variation	3
4.4. Vérification des éléments interagissant avec le tableau	3
4.5. Accéder à des commandes de vérification via la zone textuelle	3
4.6. Accéder à une aide et aux détails d'utilisation des commandes tapées textuellement	3
5. Liste des commandes	3
• Les commandes agissant directement sur le tableau	3
• Les commandes accessibles depuis la zone textuelle (voir API pour plus de détails)	4
6. Messages d'erreur	4

1. Introduction

Bienvenue dans le manuel d'utilisation de notre interface graphique pour la création d'un tableau de variation. Ce guide a pour but de vous aider à comprendre et à utiliser ce logiciel.

1.1. Objectifs et méthodes

Le principal objectif de ce manuel est de fournir une vue d'ensemble complète et des instructions détaillées pour manipuler l'interface.

1.2. Documents de référence

- Le manuel d'installation
- La documentation de l'API

2. Guide de lecture

Ce manuel est structuré de manière à vous guider de l'introduction aux fonctionnalités avancées du logiciel. Chaque section est conçue de manière indépendante afin que vous puissiez facilement trouver les informations nécessaires sans avoir à lire tout le document.

3. Concepts de base

Le logiciel inclut plusieurs concepts clés :

- La complétion du tableau de variation
- La vérification de la cohérence des éléments du tableau
- Manipuler des commandes de vérification indépendamment du tableau via une zone dédiée
- Accès à une zone dédiée à l'affichage d'informations sur les commandes textuelles

4. Mise en œuvre

4.1. Lancement du logiciel

Le logiciel étant web-based, aucune installation n'est nécessaire. (voir le manuel d'installation pour plus de détails).

4.2. L'entrée de la fonction principale

- Ajout de la fonction principale et de sa dérivée

Deux zones de texte destinées respectivement à l'entrée de la fonction principale et de sa dérivée en haut à gauche de l'interface. Attention, le manque de ces informations invalide de

la totalité du tableau de variation. Toutefois, vous êtes autorisé à compléter le tableau en dépit du manque de ces informations. L'erreur s'affiche seulement quand une commande de vérification est active.

- Nommer la fonction et la variable

4.3. Complétion du tableau de variation

- Changer les valeurs de x
- Les contenus de la ligne de la fonction dérivée
- Les variations de la fonction
- Ajouter des colonnes
- Ajouter des sous fonctions

4.4. Vérification des éléments interagissant avec le tableau

- Vérifications concernant l'entrée de la fonction
- Vérifications concernant le contenu du tableau (Dans la partie haut à droite de l'interface).
- Pour vérifier le tableau en entier, cliquez sur le bouton rouge (?) en haut à droite.

4.5. Accéder à des commandes de vérification via la zone textuelle

4.6. Accéder à une aide et aux détails d'utilisation des commandes tapées textuellement

5. Liste des commandes

- Les commandes agissant directement sur le tableau
 - validation totale
 - comparer les valeurs de x
 - valider dériv fonction
 - vérifier dériv fonction
 - les sous-fonctions

- Les commandes accessibles depuis la zone textuelle (voir API pour plus de détails)

- `init()` : Initialiser le module "tableau de variation".
- `function_definition()` : Définir la fonction principale sur laquelle on se base pour remplir le tableau de variation.
- `validate()` : Vérifier si une fonction est correctement écrite afin d'être lisible par l'API.
- `deriv()` : Retourne une dérivée de la fonction mis en argument en fonction d'une variable
- `compare_values()` : Comparer deux expressions.
- `simplify()` : Simplifier une expression en une expression équivalente sur son domaine de définition.
- `develop()` : Développer une expression arithmétique en une expression équivalente.
- `suppose_var()` : Déclarer une variable.
- `is_positive()` : Indiquer si le signe d'une expression est positif (zéro étant considéré positif).
- `print_env()` : Afficher l'environnement actuel.
- `defined()` : Indique si une expression est bien définie (y) dans l'environnement, si elle n'est pas définie (n), ou si on ne sait pas (?).
- `suppose()` : Suppose une propriété comme étant vraie dans l'environnement.
- `proof_assistant()` : Exécute des commandes comme si on les avait données à l'assistant de preuve.

6. Messages d'erreur

Si une fonction retourne une erreur, elle sera toujours sous la forme ci-dessous. (S'applique notamment dans le modules liés à l'interface avec le prouveur, le module de vérification et de validation du tableau, le module de gestion de la zone textuelle).

1.1 : erreur de syntaxe

(ex : [1.1, message, ligne, charac])

1.2 : erreur de typage.

(ex : [1.2, message, var])

1.3 : variable déjà définie..

(ex : [1.3, message])

1.4 : variable déjà initialisée.

(ex : [1.4, message])

1.5 : erreur inconsistante..

(ex : [1.5, message])

1.6 : élément DOM inexistant.

(ex : [1.6, message])

1.7 : Out of bound
(ex : [1.7, message])

1.9 : erreur interne.
(ex : [1.9, message])