

**ISEN**

ALL IS DIGITAL!

**TOULON**



**yncréa**

# **CAHIER DE CONCEPTION GÉNÉRALE**

Projet Java-Mathématiques 2017-2018

Version: CCG

Auteurs : Marion Bouteille, Thomas Artru, Yanis Meziane

ISEN Toulon - Yncrea

Maison du Numérique et de l'Innovation

Place Georges Pompidou - 83000 Toulon

Projet Mathématiques/Java – Cahier de conception générale

1/8

## Description du document

<u>Type</u>	<u>Version</u>	<u>Confidentialité</u>		
Cahier de conception générale	2.0	Usage Externe		
<u>Redacteurs</u>	<u>Nom</u>	<u>Fonction</u>	<u>Date</u>	<u>Visa</u>
	Meziane Yanis	Membre du projet	18/12/2017	
	Bouteille Marion	Chef de projet		
	Artru Thomas	Membre du projet	18/12/2017	
<u>Vérificateur</u>	Bouteille Marion	Chef de Projet	18/12/2017	
<u>Approbateur</u>	Bouteille Marion	Chef de Projet	18/12/2017	
<u>Destinataire</u>		<u>Fonction</u>	<u>Organisme</u>	
Public			ISEN	

<u>Version</u>	<u>Date</u>	<u>Redacteur</u>	<u>Modifications</u>
1.0	18/07/2016	FMC	Mise en forme
2.0	18/12/2017	Bouteille Marion Artru Thomas Meziane Yanis	Remplissage

# SOMMAIRE

<b>1 Introduction</b>	<b>4</b>
<b>2 Modules fonctionnels</b>	<b>4</b>
2.1 Architecture des Objets	4
2.2 Données utilisées par chaque objet	4
2.3 Echange de données entre objet	5
<b>3 Structures de données</b>	<b>5</b>
3.1 Définition des structures de données	5
3.2 Action portant sur les structures de données	6
3.3 Visibilité des structures de données	6
<b>4 Liste des méthodes</b>	<b>6</b>
4.1 Description des méthodes	6
4.1.1 Objet NombreComplexe	6
4.1.2 Objet Liste Nombre Complexe	7
4.1.3 Objet Affichage	7
4.1.4 Objet Liste Nombre	7
4.1.5 Objet CSV Controller	8

# 1 Introduction

Vous trouverez ici toutes les informations relatives à l'architecture du logiciel. Ainsi vous trouverez ci-dessous le découpage en objets du programme, ainsi que la définition des différentes données utilisées par le programme. Vous trouverez aussi l'arbre d'appel des méthodes ainsi que les flux de données et une liste des méthodes du programme regroupées par objets.

## 2 Modules fonctionnels

### 2.1 Architecture des Objets

Objets différents :

- Nombres Complexes
- Liste de nombres Complexes
- Liste de nombres réels
- Affichage (permet un affichage graphique)
- CSV Controller

On dispose donc de 2 modules. Un affichage, et un calcul.

### 2.2 Données utilisées par chaque objet

#### **Module Calcul:**

Dans l'objet liste de nombre Complexes:

- Un tableau de nombres complexes
- La taille du tableau

Dans l'objet nombre Complexes

- La partie réelle
- La partie imaginaire

Dans l'objet liste de nombres réels

- Un tableau de nombres réels
- La taille du tableau

Dans l'objet CSV Controller

- Un tableau de nombres complexes

### **Module Affichage:**

Dans l'objet affichage uniquement :

- Les caractéristiques de la fenêtre : Taille, visible ou non, actions, etc...

## **2.3 Echange de données entre objet**

Les objets s'échangent les données qu'ils partagent via les getters et les setters disponibles dans chaque objet.

# **3 Structures de données**

## **3.1 Définition des structures de données**

Les attributs sont tous "Private".

Objet :

### **Nombre Complexe**

Attribut :

- réel (flottant)
- imaginaire (flottant)

### **Liste Nombre Complexe**

Attribut:

- 3 Tableau (tableau de Nombre Complexe):
  - Un pour le signal d'entrée
  - Un pour la transformée de fourier
  - Un pour le signal de sortie
- taille (entier, taille du tableau)

### **Liste Nombres**

Attribut:

- 3 Tableau (tableau de Nombre réels):
  - Un pour le signal d'entrée
  - Un pour la transformée de fourier
  - Un pour le signal de sortie
- taille (entier, taille du tableau)

### CSV Controller

Attribut:

- Un tableau (tableau de Nombre complexes):

### Fenêtre Graphique

Attribut:

- La fenêtre graphique possède divers attributs nécessaires à son fonctionnement, tels que les boutons, ses panels, ses conteneurs, etc...

## 3.2 Action portant sur les structures de données

### **Tableau de Complexes**

Les méthodes de l'objet liste nombre complexe utilisent ce tableau

### **Tableau de réels**

Les méthodes de l'objet liste nombre utilisent ce tableau

## 3.3 Visibilité des structures de données

Java permet de directement paramétrer la visibilité des structures. Elles seront (par défaut) privées et accessibles via les getters et les setters.

# 4 Liste des méthodes

## 4.1 Description des méthodes

### 4.1.1 Objet NombreComplexe

Getter et setter pour les attributs détaillés en 3.1.

```
public double getReel()
```

retourne la partie réelle d'un nombre complexe

```
public double setReel(double reel)
```

```
public double getImaginaire()
```

retourne la partie imaginaire d'un nombre complexe

```
public double setImaginaire(double imaginaire)
```

```
public nombreComplexe fois(nombreComplexe nc1)  
    retourne le résultat de this * nc1.
```

```
public nombreComplexe plus(nombreComplexe nc1)  
    retourne le résultat de this + nc1.
```

```
public nombreComplexe moins(nombreComplexe nc1)  
    retourne le résultat de this - nc1.
```

```
public void affiche()  
    affiche la partie réelle et imaginaire d'un nombre complexe
```

```
public nombreComplexe conjugue()  
    retourne le conjugué de this.
```

#### 4.1.2 Objet Liste Nombre Complexe

```
public NombreComplexe getSortieFourier(int indice)  
    retourne la valeur après la transformation de Fourier
```

```
public void FTT()  
    calcul la transformée de fourier du tableau de données de l'objet et l'enregistre dans  
le tableau sortie Fourier.
```

```
public void iFTT()  
    calcul la transformée de fourier inverse à partir du tableau de sortie de Fourier et  
l'enregistre dans le tableau signal sortie.
```

#### 4.1.3 Objet Affichage

Implémente Swing. Une fonction constructeur qui crée la fenêtre:

```
public void Affichage()
```

#### 4.1.4 Objet Liste Nombre

```
public NombreComplexe getSortieFourier(int indice)
```

retourne la valeur après la transformation de Fourier

```
public void FTT()
```

calcul la transformée de fourier du tableau de données de l'objet et l'enregistre dans le tableau sortie Fourier.

```
public void iFTT()
```

calcul la transformée de fourier inverse à partir du tableau de sortie de Fourier et l'enregistre dans le tableau signal sortie.

#### 4.1.5 Objet CSV Controller

```
public void read(String adresse)
```

Permet au système d'ouvrir un fichier csv.

```
public void write(String nom)
```

Permet au système d'écrire dans un fichier csv.

## 5 Arbre des modules

