

# TP4 C++

B3129 : Pierre-Louis LEFEBVRE et Nicolas SIX

Vendredi 5 février 2016

# Table des matières

<b>I</b>	<b>Introduction</b>	<b>2</b>
<b>II</b>	<b>Cahier des charges</b>	<b>2</b>
<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Commandes</b>	<b>2</b>
2.1	Ajout d'une forme . . . . .	2
2.2	Ajout d'un segment . . . . .	2
2.3	Ajout d'un rectangle . . . . .	3
2.4	Ajout d'un polygone convexe à n côtés . . . . .	3
2.5	Ajout d'une réunion de N objets . . . . .	3
2.6	Ajout d'une intersection de N objets . . . . .	3
2.7	Suppression d'un objet . . . . .	3
2.8	Déplacement d'un objet . . . . .	3
2.9	Vérification d'appartenance d'un point à l'intérieur d'un objet . . . . .	4
2.10	Énumération des formes géométriques existantes . . . . .	4

# Première partie

## Introduction

## Deuxième partie

### Cahier des charges

#### 1 Introduction

Notre logiciel est un éditeur de formes géométriques en ligne de commande (sans interface graphique). L'éditeur permet la gestion des formes géométriques suivantes dans un plan possédant un repère cartésien  $XY$  :

- Segment de droite
- Rectangle aux côtés parallèles aux axes du graphique
- Polygone convexe
- Réunion et intersection des formes précédentes et d'autres intersections et réunions

Toutes les formes sont considérées comme *fermées* c'est à dire que les points situés sur les bords de la formes font partie de la forme. Les objets peuvent être vides de tout point (les réunions et intersections notamment).

#### 2 Commandes

##### 2.1 Ajout d'une forme

Commande

```
COMMANDE Name PARAM
```

Réponse

```
[OK|ERR]
```

Ajoute une forme de type `COMMANDE` (voir ci-dessous) à partir des paramètres `PARAM` (voir ci-dessous) et de nom `Name`, qui est un mot composé de chiffres et de lettres majuscules ou minuscules uniquement. Les paramètres de type nombre sont toujours des entiers dans la limite de `int` du `C++`, sauf indication contraire. La réponse est `OK` si la commande s'est bien exécutée, `ERR` sinon. La réponse peut s'accompagner d'un commentaire : une ligne commençant par le caractère `#` indiquant dans le cas d'une erreur le type d'erreur (`unkown command` (lorsque `COMMANDE` n'est pas une commande du logiciel), `name already used` (lorsque le nom donné à l'objet à créer est déjà utilisé pour un autre objet), `invalid parameters` (lorsque les paramètres `PARAM` est invalide)).

##### 2.2 Ajout d'un segment

Commande

```
S Name X1 Y1 X2 Y2
```

Ajoute un segment entre les points de coordonnées  $(X1, Y1)$ ,  $(X2, Y2)$ ,  $X1, Y1, X2, Y2$  étant des nombres.

## 2.3 Ajout d'un rectangle

Commande

R Name X1 Y1 X2 Y2

Ajoute un rectangle dont le segment d'extrémités de coordonnées  $(X1, Y1)$  et  $(X2, Y2)$  est une diagonale.

## 2.4 Ajout d'un polygone convexe à n côtés

Commande

PC Name X1 Y1 X2 Y2 ... Xn Yn

Ajoute un polygone de côtés de coordonnées  $(X1, Y1), (X2, Y2) \dots (Xn, Yn)$ , avec  $n \geq 3$  uniquement si ce dernier est convexe, sinon une erreur sera générée avec le commentaire non convexe polygon.

## 2.5 Ajout d'une réunion de N objets

Commande

OR Name Name1 Name2 ... NameN

Construit un objet Name comme la réunion des N objets Name1, Name2 NameN.  $N \geq 0$  (si  $N = 0$  alors on crée un objet vide). Si le nom de l'un des N objets ne correspond pas à un objet déjà existant, une erreur de paramètres invalides est renvoyée.

## 2.6 Ajout d'une intersection de N objets

Commande

OI Name Name1 Name2 ... NameN

Construit un objet Name comme l'intersection des N objets Name1, Name2 NameN.  $N \geq 0$  (si  $N = 0$  alors on crée un objet vide). Les erreurs possibles sont les mêmes que pour la réunion.

## 2.7 Suppression d'un objet

Commande

DELETE Name1 Name2 ... NameN

Réponse

[OK|ERR]

Supprime les N objets Name1, Name2 NameN. Si un nom ne correspond pas à un objet existant, aucun objet n'est supprimé et une erreur de paramètres invalides est renvoyée.

## 2.8 Déplacement d'un objet

Commande

MOVE Name dX dY

Réponse

[OK|ERR]

Déplace l'objet Name d'une distance dX (nombre) sur l'axe des X et dY (nombre) sur l'axe des Y

## 2.9 Vérification d'appartenance d'un point à l'intérieur d'un objet

Commande

HIT Name X Y

Réponse

[YES|NO]

Renvoie YES si le point de coordonnées (X, Y) se trouve à l'intérieur (ou au bord) de l'objet Name NO sinon.

## 2.10 Énumération des formes géométriques existantes

Commande

LIST

Réponse

Desc1  
Desc2  
...  
DescN

Affiche les descripteurs des formes géométriques existantes, selon le format suivant :

Pour le rectangle, le segment et le polygone, la description est identique à la commande d'ajout correspondante.

**Exemple :** Le rectangle de nom Name1 et de côtés opposés (2,3) et (3,8) :

R Name1 2 3 3 8

Pour l'intersection et la réunion de N objets, la forme est la suivante. Les descripteurs complets de l'ensemble des N objets (y compris ceux d'éventuels autres intersections ou réunion), sauf que leur nom est le nom d'origine immédiatement suivi sans séparateur d'un tiret (-) et du nom de l'intersection ou de la réunion. Suivi de la commande de création de l'intersection ou de la réunion, et enfin de la commande de destruction des N objets.

**Exemple :** La réunion OR1 du rectangle R1, du segment S1 et de l'intersection OI1 du polygone P1 et du segment S2 :

```

R R1-OR1 1 2 3 8
S S1-OR1 1 7 45 48
PC P1-OI1-OR1 14 45 48 12 45 78 89 56 78 10
S S2-OI1-OR1 0 0 2 2
OI OI1-OR1 P1-OI1-OR1 S2-OI1-OR1
DELETE P1-OI1-OR1
DELETE S2-OI1-OR1
OR OR1 R1-OR1 S1-OR1 OI1-OR1
DELETE R1-OR1
DELETE S1-OR1
DELETE OI1-OR1

```

La persistance d'un ensemble d'objets construit de cette manière; le fichier de sauvegarde est un fichier de format texte.