

# Cahier des charges TP Héritage

ESPEUTE Clément, GAO Yiqin

15 janvier 2016

## 1 Présentation

Le but du programme est de réaliser un éditeur de formes géométriques et permettant de manipuler celles-ci. Il doit permettre la gestion de :

- Segments
- Rectangles
- Polygones convexes

On peut effectuer les manipulations suivantes :

- Ajout d'un nouvel objet
- Intersection ou réunion de formes déjà créés
- Vérifier qu'un point est dans un objet
- Suppression et déplacement d'un objet
- Sauvegarde et Chargement des figures créés dans un fichier au format .txt

## 2 Commandes

Le programme doit gérer les commandes suivantes.

## 2.1 Ajouter un segment

Commande : S Name X1 Y1 X2 Y2

Réponse : [OK|ERR]

Description : Ajoute un segment entre les points  $(X_1, Y_1)$ ,  $(X_2, Y_2)$ . L'objet à un nom [name] qui est un mot composé de lettres et ou chiffres sans séparateurs à l'intérieur. Renvoie OK si la commande s'est bien exécutée, ERR dans le cas contraire (Exemple paramètres invalides ou erreur mémoire). La réponse peut s'accompagner d'un commentaire (ligne qui commence avec un #)

## 2.2 Ajouter un rectangle

Commande : R Name X1 Y1 X2 Y2

Réponse : [OK|ERR]

Description : Ajoute un rectangle défini par son point gauche-haut  $(X_1, Y_1)$ , et son point droite-bas  $(X_2, Y_2)$ . L'objet à un nom [name] qui est un mot composé de lettres et ou chiffres sans séparateurs à l'intérieur. Renvoie OK si la commande s'est bien exécutée, ERR dans le cas contraire (Exemple paramètres invalides ou erreur mémoire). La réponse peut s'accompagner d'un commentaire (ligne qui commence avec un #)

## 2.3 Ajouter un polygone convexe

Commande : PC Name X1 Y1 X2 Y2 ... Xn Yn

Réponse : [OK|ERR]

Description : Ajoute un polygone défini par les points :  $(X_1, Y_1)$ ,  $(X_2, Y_2)$ , ...,  $(X_n, Y_n)$ . avec  $n \geq 3$ . Renvoie OK si le polygone a bien été généré. Si le polygone n'est pas convexe, une erreur est générée (ERR dans la sortie standard). Le dernier point sera relié au premier point.

## 2.4 Opération de réunion

Commande : OR Name Name1 Name2 ... NameN

Réponse : [OK|ERR]

Description : Crée un nouvel objet name qui est la réunion de copies des objets nommées Name1 ... NameN. Renvoie OK si la création s'est bien passée, sinon renvoie ERR si les arguments sont invalides (exemple nom erroné).

## 2.5 Opération de intersection

Commande : OI Name Name1 Name2 ... NameN

Réponse : [OK|ERR]

Description : Crée un nouvel objet name qui est l'intersection de copies des objets nommées Name1 ... NameN. Renvoie OK si la création s'est bien passée, sinon renvoie ERR si les arguments sont invalides (exemple nom erroné).

## 2.6 Opération d'appartenance

Commande : HIT Name X Y

Réponse : [YES|NO]

Description : Vérifie si le point de coordonnées X Y est à l'intérieur (y compris sur le périmètre avec  $\pm 1$  unité de marge d'erreur sur l'appartenance au segment ) de l'objet Name. Renvoie YES si c'est le cas, sinon NO.

## 2.7 Opération de suppression

Commande : DELETE Name1 Name2 ... NameN

Réponse : [OK|ERR]

Description : Supprime les objets données en paramètres. Renvoie OK si tous les objets ont été supprimés, renvoie ERR si un nom est invalide (aucun objet supprimé dans ce cas)

## 2.8 Opération de Déplacement

Commande : MOVE Name dX dY

Réponse : [OK|ERR]

Description : Déplace l'objet name de dX sur l'axe horizontal et de dY sur l'axe Vertical. Renvoie OK quand tout s'est bien passé, ERR si paramètre invalide.

## 2.9 Opération d'Enumération

Commande : LIST

Réponse : Voir Description

Description : Liste tout les objets existants sous la forme Nom : Type, Description

Les descriptions sont :

Segment : X1, Y1, X2, Y2

**Rectangle** : X1, Y1, X2, Y2

**Polynome Convexe** : X1, Y1, ... Xn, Yn

**Réunion** : On fait une tabulation et on décrit les objets dans la réunion sous la forme **Type**, **Description** séparés par un retour à la ligne. Exemple d'une Réunion composée d'un Rectangle, d'un segment et d'une autre réunion (qui contient un rectangle)

```
Reunion1 : Reunion
    CopyRectangle1 : Rectangle, 10, 10, 50, 50
    CopieSegment1 : Segment, 10, 10, 40, 40
    CopieReunion2 : Reunion
        Rectangle, 50, 50, 100, 100
Autre Description d'objet
```

**Intersection** : Idem que pour la réunion

## 2.10 Opération d'Annulation

Commande : UNDO

Réponse : [OK|ERR]

Description : Annule la dernière commande effectuée (mis à part UNDO et REDO). Renvoie OK si l'annulation a bien eu lieu, et ERR si jamais on ne peut plus annuler. Le programme stocke jusqu'à 20 niveau d'annulation.

## 2.11 Opération de Refaire

Commande : REDO

Réponse : [OK|ERR]

Description : Refais la dernière commande annulée. Ne peut être appelé que si il y a déjà eu un REDO et qu'aucune autre commande a été effectuée entre temps. Renvoie OK si la reprise s'est bien passée, sinon renvoie ERR.

## 2.12 Opération de chargement

Commande : LOAD filename

Réponse : [OK|ERR]

Description : Charge un ensemble d'objet à partir d'un fichier texte. Les objets sont ajoutés au dessin courant. Renvoie OK si les objets ont bien été créés, et renvoie ERR si un objet n'a pas pu être créé car son nom existe déjà.

### **2.13 Opération de sauvegarde**

Commande : `SAVE filename`

Description : Sauvegarde l'ensemble des commandes nécessaires à la création de la figure courante dans le fichier `filename`. Renvoie OK si la sauvegarde à bien eu lieu, renvoie ERR si la sauvegarde à échouée.

### **2.14 Opération de remise à zéro**

Commande : `CLEAR`

Description : Supprime tous les objets courants. Renvoie toujours OK

Réponse : OK

### **2.15 Opération de fermeture**

Commande : `EXIT` Quitte le programme.