

## TP Algorithmique N° 3 Gestion de fichiers

L'objectif de ce TP est de stocker vos objets géométriques (des polygones 2D) dans des fichiers de façon à pouvoir les réutiliser.

Bien entendu, vous utiliserez les fonctions définies dans les bibliothèques du précédent TP pour créer les polygones.

### I) Enregistrement et lecture de fichiers

En plus des fonctions existantes (`afficherPolygone`, `translationPolygone`, ... ), vous créerez les fonctions suivantes :

```
void enregistrerPolygone( const char &nomFichier, const polygone &p );  
void lirePolygone( const char &nomFichier, polygone &p );
```

#### Enregistrement d'un polygone

L'enregistrement d'un polygone s'effectuera en écrivant les coordonnées x et y de chaque sommet le constituant, en texte "clair" (ASCII). Les fonctions à employer sont `fopen()`, `fprintf()` et `fclose()`. Elles sont déclarées dans les fichiers d'en-tête de la bibliothèque standard du C, `stdlib.h` et `stdio.h`.

► ajouter : `#include <stdlib.h>` et `#include <stdio.h>` dans vos fichiers source.

Exemple de fichier obtenu (polygone composé de 7 sommets) :

```
1.0 2.0  
1.5 1.0  
4.0 1.5  
5.0 2.5  
3.0 4.0  
1.2 3.0  
1.0 2.0
```

Utiliser un nom du fichier avec une extension `.dat`.

Ex : "poly1.dat".

#### Lecture d'un polygone

Fonctions à utiliser : `fopen()`, `feof()`, `fscanf()` et `fclose()`.

La fonction `lirePolygone()` devra lire ligne par ligne (sommet par sommet) le fichier donné en paramètre et stocker les valeurs lues avec `fscanf()` dans le tableau de sommets.

Ne pas oublier de compter les sommets.

## II) Affichage des polygones avec le programme *gnuplot*

**gnuplot** est un programme qui permet de visualiser des données (fonctions mathématiques ou tableaux de valeurs).

Nous utiliserons les fichiers que vous avez créés comme tableau de valeurs en entrée de **gnuplot** qui reliera graphiquement les sommets par des segments.

### ► Lancer **gnuplot**

Pour afficher le polygone stocké dans le fichier "poly.dat", taper la commande **gnuplot** :

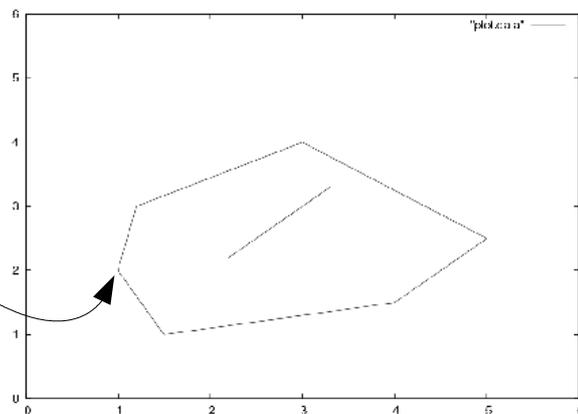
► `plot "poly.dat" with lines`

Pour séparer deux sommets, il suffit d'insérer une ligne vide dans la liste.

Essayer avec le fichier composé comme suit :

```
1.0 2.0
1.5 1.0
4.0 1.5
5.0 2.5
3.0 4.0
1.2 3.0
1.0 2.0

2.2 2.2
3.3 3.3
```



et la commande **gnuplot** : `plot [0:6] [0:6] "poly.dat" with lines`

Ceci vous permettra d'afficher plusieurs polygones dans une même image simplement en concaténant les fichiers de polygones.

### **Travail à effectuer :**

Étant donné un polygone, lui appliquer une translation puis une rotation puis une homothétie et afficher toutes les étapes intermédiaires graphiquement.

Il suffit pour cela, à chaque étape, d'enregistrer le polygone transformé à la fin d'un fichier.

Pour ajouter des données dans un fichier il faut ouvrir en mode "ajout" (append).

Ex :

```
FILE h = fopen( "scene.dat", "a" );
```