

Rapport de Projet Ada

Thomas Conte

Cyril Gaunet

1 Introduction

Pour commencer, nous avons choisi d'organiser notre programme de la manière suivante :

- Un fichier `boites.adb` contenant le corps principal du programme,
- Un paquetage `Demi_Boite` permettant de tracer les différentes facettes d'une demi-boîte,
- Un paquetage `Cotes` permettant de tracer chaque côté de chaque facette,
- Un paquetage `Svg` permettant de générer le code *svg* correspondant aux différents polygones,
- Un dernier paquetage `Utils` utilisé par le `boites.adb` contenant des fonctions utiles pour rendre le programme principal plus clair.

Les noms des variables utilisés sont assez explicites la plupart du temps, hormis les variables correspondantes aux paramètres et qui sont nommées selon celui auquel elles correspondent. De plus, nous avons décidé d'appeler `P`, `Px` ou `Py` la **p**lace entre le début du côté et la première queue ou encoche. Cette place est donc toujours supérieure ou égale à `T`.

2 Le programme boites

Ce fichier contient la définition des grandeurs de la boîte demandée. C'est pourquoi ce programme commence par lire les arguments de la ligne de commande afin de s'assurer que la demande est correcte. On en déduit les dimensions de la demi-boîte interne puis les dimensions du fichier *svg*. Ensuite, il suffit de créer ou d'ouvrir le fichier demandé selon s'il existe déjà ou pas, puis on appelle la fonction `Tracer_Demi_Boite` une fois pour chaque demi-boîte, on termine le fichier *svg* et on le ferme. On obtient ainsi un programme principal clair et concis grâce à la séparation en différents paquetage.

3 Le paquetage Demi_Boite

Ce paquetage contient la procédure `Tracer_Demi_Boite`, seule fonction appelée par le programme principal et consiste uniquement à appeler cinq fois la fonction `Tracer_Facette` avec des paramètres différents.

La procédure `Tracer_Facette` quant à elle consiste à appeler les procédures de

Cotes correspondantes à la face passée en parametre. Afin de passer les bons paramètres à ces procédures, il faut d'abord calculer `Nx`, `Ny`, `Px` et `Py` grâce à la fonction `CalculNP`.

4 Le paquetage Cotes

Il contient dix procédures différentes mais très similaires, une pour chaque côté de chaque facette, qui décrivent la manière dont vont s'enchaîner les queues et les encoches et qui ajoutent des points dans la balise `polygon` grâce à la procédure `PutPoint`.

Cette dernière a été créée dans le but de réduire le nombre d'appels aux différentes procédures `Put` de `Ada.Text_IO` et `Ada.Float_Text_IO`.

Il contient aussi une redéfinition de l'opérateur `"*"` permettant de multiplier un entier avec un flottant, ce qui nous évite de convertir les entiers en flottants de manière répétée.

Nous aurions pu choisir de créer les côtés par symétrie ou en utilisant des boucles mais nous avons préféré créer des procédures de manière exhaustive pour permettre une meilleure lisibilité du code et une plus grande liberté dans la création de chaque côté.

5 Le paquetage Svg

Il permet de générer tous les bouts de code *svg* grâce à ses différentes procédures : `Svg_Header` pour l'en-tête, `Svg_Footer` pour le pied, `Svg_BeginPolygone` pour l'ouverture de la balise `polygon` et `Svg_EndPolygone` pour la fermeture. Il est aurait peut-être été plus logique de placer la procédure `PutPoint` dans ce paquetage mais étant donné qu'elle utilise une variable globale du paquetage `Cotes`, nous avons choisi de procéder ainsi pour une syntaxe plus allégée.

6 Le paquetage Utils

Dans ce paquetage, la fonction `Exist` renvoie `True` si le fichier passé en paramètre existe et `False` sinon. Pour le savoir, il essaye de l'ouvrir comme s'il existait. Si une exception est levée, c'est que le fichier spécifié n'existe pas, donc on renvoie `False`. Sinon, on ferme le fichier puis on renvoie `True`.

Ensuite, la fonction `Max` renvoie le maximum de deux nombres flottants, nous l'avons rédigée uniquement pour que le programme principal soit plus clair.

Pour finir, la procédure `Read_Args` reçoit en argument toutes les variables qui sont censées être spécifiées dans la ligne de commande. Elle lit tous les arguments et lève éventuellement une exception si l'un d'entre eux n'est pas précisé ou si l'une des valeurs données ne permet pas le dessin correct d'une boîte.