

Min Projet : INFO S3 2017

Date : 11 janvier 2017

Par Sidi EHMETY

Pré-requis : aucun

Code : INF3070

Crédits : 3(0, 0, 39)

Objectif :

Mise en pratique de connaissances acquises dans le cadre d'un projet encadré initiation des étudiants au travail en équipe, coordination, échanges. Initiation aux différentes étapes d'un projet, conduite de projet, respect de délais d'impentation, validation et présentation

Délais de réalisation : 3 semaines

Sujet :

Ce mini projet s'intéresse à la représentation des automates fini en Ocaml. On vous propose la structure suivante pour représenter des automates :

type automate = {i:int; t: (int*char*int)list; e:int list; f:int list};;

Veillez noter que cette définition on représente les ensembles par des listes. Elle n'est pas obligatoire. Vous pouvez faire votre propre définition si vous le souhaitez.

On vous demande d'écrire en Ocaml les fonctions suivantes (partie obligatoire du mini projet) :

- 1) Fonction « **complet** » qui prend un automate et retourne **true** si cette automate est complet sinon elle retourne **false**.
- 2) Fonction « **deterministe** » qui prend un automate et retourne **true** ou **false** selon que l'automate est déterministe ou non.

Pour les deux fonctions suivantes, vous êtes demandés de programmer l'une d'elles au choix.

- 3) Une fonction « **dessiner_automate** : » qui prend un automate et dessine cet automate sur une fenêtre graphique.
- 4) Une fonction « **minimaliser** » qui prend un automate quelconque et retourne un automate minimal.

Le livrable, à fournir

Dans tous les cas il faudra donner, un tout cohérent avec :

- Les codes sources et les inclus (corrects: compilables et bien commentés).
- Des exemples d'exécution : les données en entrée, et celles en sortie qui ont été obtenues, avec ces mêmes sources
- un document qui explique ce que fait le programme, comment l'utiliser, et quelle est sa conception. Donner un exemple qui utilise les mêmes données que celles qui auront été fournies (en entrée et en sortie).

Bibliothèque graphique

Pour dessiner les images fractales, nous allons utiliser la bibliothèque graphique Graphics prédéfinie dans Ocaml. Pour utiliser cette bibliothèque dans l'interpréteur, vous devez taper la commande suivante :

ocaml graphics.cma

puis taper tout de suite la commande open Graphics. Pour compiler votre programme, vous devez exécuter le compilateur avec la commande suivante :

ocamlc -o monprogramme graphics.cma monprogramme.ml

Pour utiliser cette bibliothèque graphique, vous devez tout d'abord ouvrir une fenêtre graphique à l'aide de la fonction

open_graph : string -> unit.

Par exemple, l'appel open_graph " 300x100" ouvre une fenetre graphique de 300 pixels de large et 100 pixels de haut. Le pixel en (0; 0) étant situé en bas à gauche.

Pour tracer des lignes, on utilisera les fonctions

moveto : int -> int -> unit et lineto : int ->int -> unit

qui permettent respectivement de positionner le point courant et de tracer une ligne du point courant vers le point passe en argument, ce point devenant ensuite le point courant.

En utilisant les fonctions précédentes, écrire une fonction tracer

ligne : (float * float) -> (float * float) -> unit

telle que tracer ligne (ax,ay) (bx,by) trace le segment entre les points (ax,ay) et (bx,by).

Automate Minimal :

Voir l'Algorithme de Moore de minimisation d'un automate fini sur wikipedia.