



ÉPREUVE ÉCRITE	Branche : Informatique / Partie pratique
Section : B	N° d'ordre du candidat :
Date de l'épreuve : 20 septembre 2016	Durée de l'épreuve : 80 minutes / 30 points

Le but de l'épreuve est d'implémenter dans Delphi l'*algorithme de Chaikin* (1908-1992), permettant d'« arrondir » une ligne polygonale.

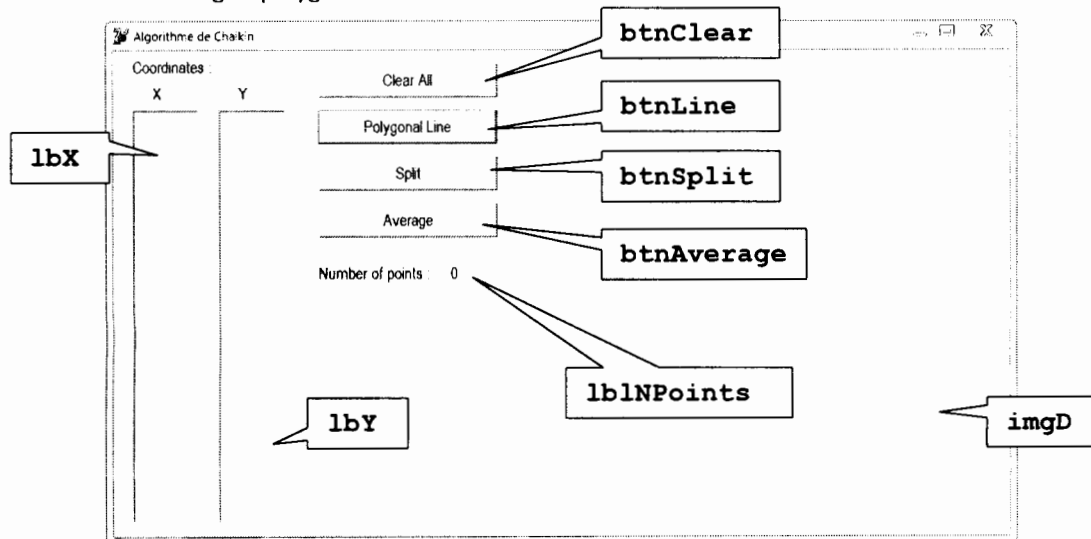


fig. 1

- (1) Créer l'interface graphique de la figure 1 en respectant les dénominations indiquées. Les dimensions de l'image peuvent être choisies librement, mais on travaillera avec les propriétés **Width** et **Height** de l'image dans le programme. Le libellé **lbNPoints** sera initialisé à 0. (4 points)
- (2) Au démarrage du programme, le canevas de l'image se présentera avec un fond blanc, sans bord, comme sur la figure 1. Un clic sur le bouton **Clear All** permet de réinitialiser tous les composants du formulaire. (4 points)
- (3) Au début de l'exécution, l'utilisateur cliquera un certain nombre de fois avec le **bouton gauche** de la souris sur le canevas de l'image. Un clic sur le pixel de coordonnées (x, y) fera apparaître un cercle noir **transparent** de centre (x, y) et de diamètre 7 pixels. L'abscisse x et l'ordonnée y seront ajoutées aux listes **lbX** et **lbY** respectivement. Le libellé **lbNPoints** devra toujours afficher le nombre correct de points (c.-à-d. de cercles) représentés sur l'image. (4 points)
- (4) Ecrire une procédure **line** qui prend en entrée les deux listes **lbX** et **lbY** ainsi que l'image **imgD** et qui a) efface l'image dessinée auparavant (mais pas les listes !) et b) dessine en noir la ligne polygonale reliant les points successifs dont les coordonnées se trouvent dans les listes **lbX** et **lbY** respectivement. Plus précisément, si $lbX = (x_1, x_2, \dots, x_n)$ et $lbY = (y_1, y_2, \dots, y_n)$, on dessinera les cercles **transparents** de centres $C_1(x_1, y_1)$, $C_2(x_2, y_2)$, ..., $C_n(x_n, y_n)$, tous de diamètre 7 pixels, et on tracera tous les segments $[C_i, C_{i+1}]$. Un clic sur le bouton **Polygonal Line** appellera la procédure **line** afin de faire apparaître la ligne polygonale. (voir fig. 2) (6 points)

- (5) Un clic sur le bouton **Split** aura comme effet de remplacer la ligne polygonale $(C_1, C_2, C_3, \dots, C_n)$ par la ligne polygonale $(C_1, M_1, C_2, M_2, C_3, \dots, M_{n-1}, C_n)$ où M_i est le milieu de $[C_i, C_{i+1}]$. Les coordonnées des milieux seront d'abord insérés aux endroits corrects dans les listes **lbX** et **lbY**, puis on appellera la procédure **line** pour dessiner la nouvelle ligne polygonale. Le nombre de points représentés devra être correctement actualisé (voir fig. 2 et 3). (6 points)

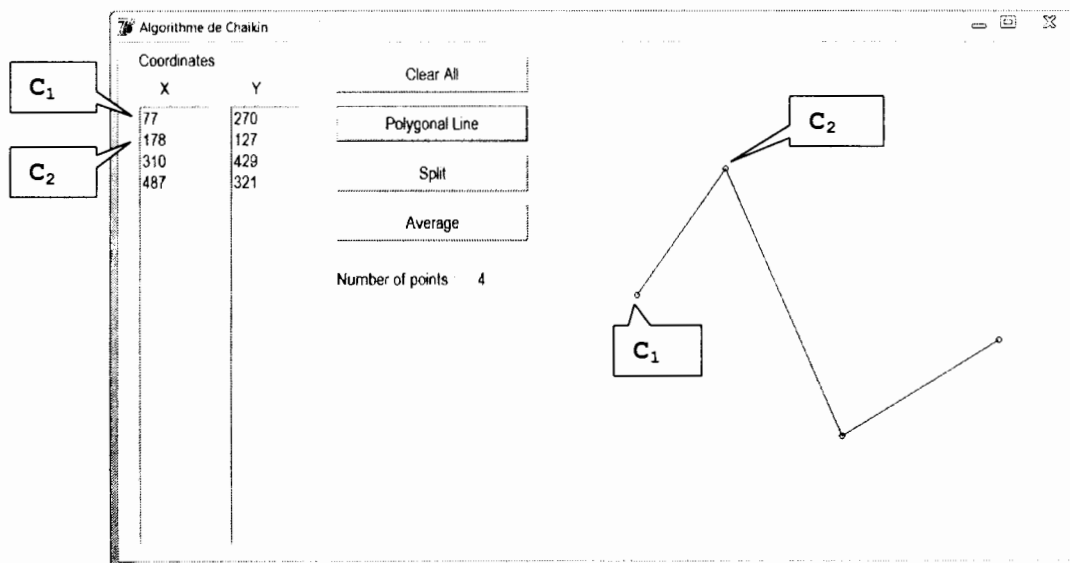


fig. 2 : après un clic sur **Polygonal Line**

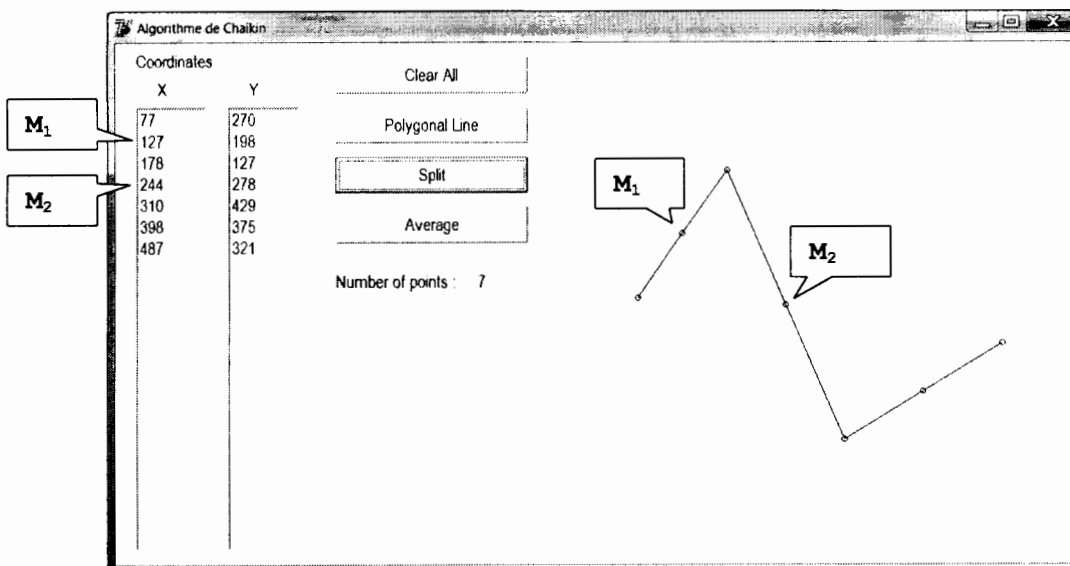


fig. 3 : après un clic sur **Split**

- (6) Un clic sur le bouton **Average** aura comme effet de remplacer la ligne polygonale $(P_1, P_2, P_3, \dots, P_n)$ par la ligne polygonale $(Q_1, Q_2, Q_3, \dots, Q_{n-1})$ où Q_i est le milieu de $[P_i, P_{i+1}]$. En reprenant l'exemple d'exécution de la figure 3, on a $P_1(77, 270)$, $P_2(127, 198)$, donc $Q_1(102, 234)$: c'est le premier point sur la figure 4. Les coordonnées des points Q_i seront d'abord écrits dans les listes **lbX** et **lbY**, puis on appellera la procédure **line** pour dessiner la nouvelle ligne polygonale. Le nombre de points représentés devra être correctement actualisé (voir fig. 4). (6 points)

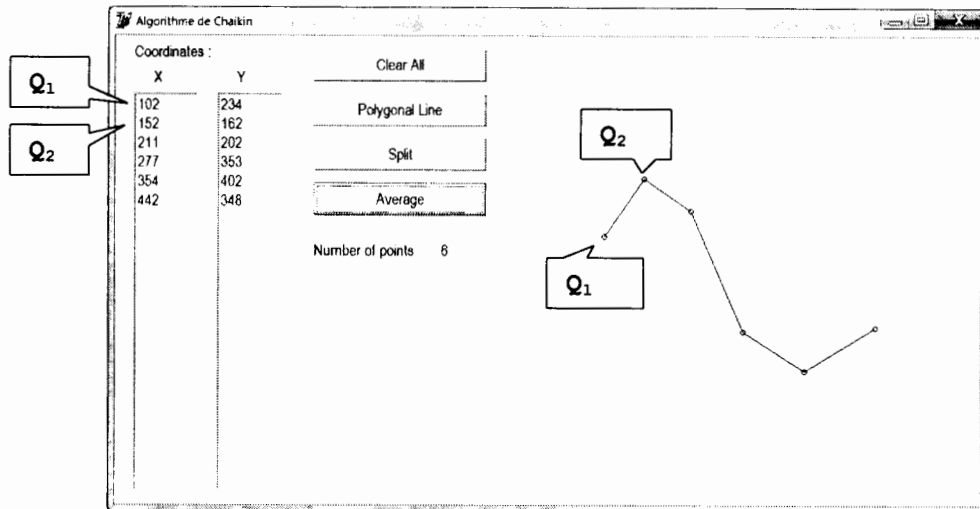


fig. 4 : après un clic sur **Average**

Remarque : En répétant plusieurs fois de suite les opérations **Split** et **Average** dans cet ordre, la ligne polygonale s'arrondit de plus en plus. (voir fig. 5)

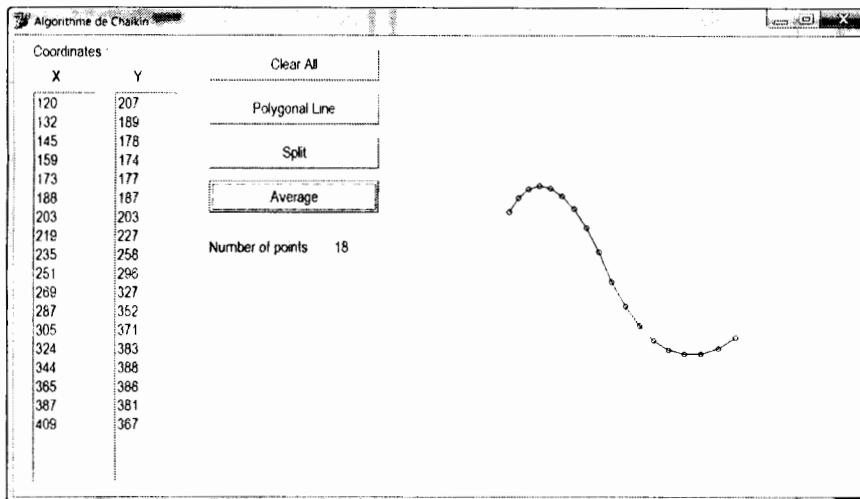


fig. 5 : après 2 autres répétitions des opérations **Split/Average**