

Infographie
Travaux pratiques
Feuille 3 (dessin 2D: le remplissage de polygones)

Rappels:

- Compilez le programme JAVA :
javac dessin2D.java
- Lancez l'applet JAVA :
appletviewer dessin2D.html
- Aide JAVA en ligne :
 - Google : java api 1.6
 - ou : <http://download.oracle.com/javase/6/docs/api/>

Exercice 1. Finir les exercices non optionnels des feuilles 1 et 2.

Exercice 2. * Remplissage d'un polygone.

Écrire une méthode permettant de remplir un polygone dont les sommets ont été préalablement saisis à la souris. Ajuster l'interface graphique pour que le polygone soit tracé au moment de la saisie des points et soit rempli quand on appuie sur le bouton "Remplissage du polygone".

Exercice 3. Suite de l'exercice 3 de la feuille 2

Modifier l'applet *dessin2D* de manière à pouvoir faire le remplissage d'un cercle.

Exercice 4.

Le but de cet exercice est de pouvoir déplacer un sommet d'un polygone. Pour cela, on veut que lorsque l'utilisateur presse la souris, l'applet sélectionne le sommet du polygone le plus proche du point où l'utilisateur a pressé la souris, et attribue à ce sommet les coordonnées du point où l'utilisateur lâche la souris.

- (1) Rajouter une ligne *modifie* dans la liste modifiable de *dessin2D*.
- (2) Écrire une méthode *int selectionne(Vector poly, Point A)* qui, pour un *Vector* contenant des objets de la classe *Point*, renvoie l'indice de l'élément de *poly* le plus proche de *A*.
- (3) Ajuster les actions de souris de manière à ce que, lorsque l'utilisateur presse sur la souris (*modifie* étant sélectionné dans le menu déroulant), l'applet modifie *poly* en remplaçant le point de *poly* le plus proche de l'endroit où l'utilisateur a pressé la souris, pour lui attribuer les coordonnées du point où l'utilisateur a lâché le bouton de la souris.
- (4) Faire en sorte que le nouveau polygone soit tracé et que l'on puisse le remplir ou le fenêtrer.

Exercice 5.

Utiliser la méthode *selectionne* de l'exercice précédent pour faire en sorte que l'utilisateur puisse enlever un sommet d'un polygone. L'applet enlèvera le sommet le plus proche du point où l'utilisateur presse la souris.

Exercice 6.

L'objectif de cet exercice est de pouvoir rajouter un sommet à un polygone. Si on note P la suite des sommets d'un polygone, il faut déterminer le sommet i après lequel l'applet insérera le nouveau sommet P' . Pour cela, on veut déterminer l'arête $[P[i], P[i + 1]]$ qui est la plus proche de P' .

Vous admettez que la distance entre un point C et une arête $[A, B]$ est

$$|CM_{t_0}|,$$

avec

$$M_t = tA + (1 - t)B$$

et

$$t_0 = \begin{cases} 0 & , \text{ si } \frac{\overrightarrow{BC} \cdot \overrightarrow{BA}}{\overrightarrow{BA} \cdot \overrightarrow{BA}} < 0 \\ \frac{\overrightarrow{BC} \cdot \overrightarrow{BA}}{\overrightarrow{BA} \cdot \overrightarrow{BA}} & , \text{ si } 0 \leq \frac{\overrightarrow{BC} \cdot \overrightarrow{BA}}{\overrightarrow{BA} \cdot \overrightarrow{BA}} \leq 1 \\ 1 & , \text{ si } 1 < \frac{\overrightarrow{BC} \cdot \overrightarrow{BA}}{\overrightarrow{BA} \cdot \overrightarrow{BA}} \end{cases}$$

On rappelle que

$$\overrightarrow{(U_x, U_y)} \cdot \overrightarrow{(V_x, V_y)} = U_x V_x + U_y V_y.$$

- (1) Écrire une méthode *float determineDistance(Point A, Point B, Point C)* qui détermine la distance entre l'arête $[A, B]$ et le point C .
- (2) Écrire une méthode *int determineIndiceInsertion(Vector poly, Point A)* qui détermine l'indice i tel que la distance entre l'arête $[poly[i], poly[i + 1]]$ et A soit la plus faible possible.
- (3) Ajuster l'interface graphique pour que l'utilisateur puisse rajouter un point saisi à la souris dans un polygone. (Ce point doit être inséré après le sommet déterminé par *determineIndiceInsertion*.)

Exercice 7.

Imaginer et rajouter des fonctionnalités à votre applet *dessin2D* (translation d'un polygone à la souris, remplissage avec des hachures, couleurs, copie d'un polygone, ...).