**COMPUTER GRAPHICS AND ANIMATION**

**PROGRAMMING ASSIGNMENT 7**

**Drawing Polygons – Linked List**

**Difficulty: 2.5/5**

Create an application that allows the user to draw polygons and polylines on a screen. The application should be able to do the following:

* Draw a polygon on the screen.
* Draw a polyline on the screen.
* Delete a polygon/polyline.
* Add a vertex to a polygon/polyline. The user can select which existing vertices will be adjacent to the added vertex.
* Edit the location of a vertex on a polygon/polyline.
* Delete a vertex from a polygon/polyline. The user can select which vertex to delete.
* Change the colour of a polygon.
* Clear the screen. This deletes all polygons and polylines.
* Save/load data of polygons and polylines to/from a file.
* Determine whether a polygon is convex or not.
* Calculate the perimeter and area of a polygon.
* Bonus points for user friendliness. Negative points for extreme user unfriendliness.

The polygons, polylines, and collection of polygons/polylines are represented using linked lists. You **may not** use the built-in linked list class provided by the programming language.

You may use the built-in procedure for drawing a line.

What to submit:

* An executable file (EXE)
* The source code (+ modules or libraries if necessary)
* A report.

The report should contain explanations of at least the following:

1. Introduction.

* What is the program about?
* In what language is the program implemented?

1. Basic theory.

* Explain the basic properties of a polygon and polyline.
* Explain how to calculate the perimeter and area of a polygon (if implemented).
* Explain how to determine whether a polygon is convex (if implemented).
* Explain how to represent a polygon/polyline using a linked list.
* Explain vertex processing on a polygon.
  + What is vertex processing?
  + Give some examples of vertex processing.
  + Write the general pseudocode for vertex processing.
* Explain edge processing on a polygon.
  + What is edge processing?
  + Give some examples of edge processing.
  + Write the general pseudocode for edge processing.

1. Implementation

* Explain the main interface of the program.
* Explain every feature in the program and how to use them.

1. Design

* Explain the main data structures (if any) used in the program.
  + How are the points represented in the program?
  + How are the polygons/polylines represented in the program?
  + How is the collection of polygons/polylines represented in the program?
* Explain the main/global variables used in the program.
* Explain how the bonuses (if done) are implemented.

1. Evaluation

* Evaluation of the main features (drawing polygons and polylines). Try the following test cases:
  + Add a polygon/polyline.
  + Delete a polygon/polyline.
  + Adding a vertex to an existing polygon/polyline.
    - Add a vertex as the first vertex.
    - Add a vertex as a “middle” vertex.
    - Add a vertex as the last vertex.
  + Changing the location of a vertex on an existing polygon/polyline.
  + Deleting a vertex from a polygon.
    - Delete the first vertex.
    - Delete a “middle” vertex.
    - Delete the last vertex.
* Also perform a test case for all the bonuses you implemented.
* Include screenshots of each test case.
* Explain whether each case is successful.

1. Work log.

* Record the date and time of every moment you work on this assignment and job description of each member at each session. The work log should be a table with the following columns:
  + Date
  + Activity / progress
  + Personnel involved
* Write a summary of the implementation of each requirement given in the first page. For each requirement, explain whether that requirement is fully implemented, partially implemented, or not implemented at all. Give explanations if necessary.

1. Conclusion and remarks.

* Does the program work as expected?
* If some parts of the program do not work as expected, explain why.
* What are your comments about this assignment?

You must report your progress every week. The following features must be implemented during your progress report:

Week 1

* The program allows the user to add a polygon on the screen. For now, just one polygon on the screen is enough.
* The program allows the user to add a vertex on the polygon.
* The program allows the user to delete a vertex from the polygon.
* The program allows the user to edit the location of a vertex.
* The program allows the user to change the color of the polygon.
* The program allows the user to clear the screen.

Week 2

* The program allows the user to draw multiple polygons on the screen.
* The program allows the user to delete a polygon from the screen.
* The program allows the user to add, edit, and delete a vertex from a polygon.
* The program allows the user to save the polygons to a file.
* The program allows the user to load polygons from a file.

Submit the assignment no later than midnight, 16 February 2020, to [x60880@yahoo.com](mailto:x60880@yahoo.com).

You must do a presentation of your work no later than 21 February 2020.