Assignment 1

Rajan Khade - MT2024077

Instructor: Prof. Jaya & Prof. Srikanth

Date: February 12, 2025

Code Implementation

Important concepts used in implementation involve following things:

- Ear Clipping for Triangulation (Library: Earcut)
- Beaking non-convex self intersecting polygon into simple polygons (library: JavaScript CLipper)

Challenges Faced:

- Finding solution to triangulate non-convex self intersecting polygons

The approach I ended up using is converting them into simple polygons and then triangulating by ear clipping

Questions and Answers

When we perform transformations on the scene, we treat the scene as a grouping of all the shapes as a single entity. How would you implement grouping and ungrouping shapes if the user would like to choose which shapes will be treated together as a group for transformations?

Representation of Groups

- Group ID-Based Approach:

Each shape in the scene has an attribute representing its group. By default, this attribute is empty, indicating that the shape is independent. When a user selects multiple shapes to form a group, a unique identifier is assigned to all selected shapes. The transformation system then processes all shapes with the same group identifier together.

- Explicit Groups List Approach:

Instead of assigning group information to each shape, a separate list of groups is maintained. Each group entry stores the identifiers of the shapes that belong to it. When a transformation is applied, the system looks up the group in this list and processes the corresponding shapes collectively.

Applying Transformations to a Group

Once a group is defined, transformations should be applied relative to a common reference point. This is usually the **centroid** of the group, computed as the average position of all shapes in the group.

- **Translation:** Each shape in the group is moved by the same offset.
- **Rotation:** Shapes are rotated around the group's centroid rather than their individual centers.
- **Scaling:** The distance of each shape from the centroid is adjusted proportionally.

This ensures that transformations maintain the spatial relationships between the shapes in the group.

Ungrouping Mechanism

Ungrouping allows the shapes to become independent again.

- In the **Group ID-Based Approach**, ungrouping resets the group attribute of all shapes in the group.
- In the **Groups List Approach**, removing a group entry disassociates the shapes, allowing them to be manipulated separately.

Why is the use of centroid important in transforming a primitive or a group of primitives? (Hint: transformations such as rotation and scaling.)

The **centroid**, which represents the geometric center of a shape or a group of shapes, serves as a natural and mathematically stable reference point. Using the centroid ensures that transformations behave predictably and preserve the spatial relationships between individual elements.

Role of the Centroid in Rotation

Rotation is performed about a fixed point, and choosing an appropriate pivot is critical to maintaining the intended structure of the shape or group.

- **Rotation About the Centroid:**
  When rotating around the centroid, all points move in circular paths relative to the centroid. This keeps the shape's spatial distribution intact without causing distortion or displacement.
- **Rotation About Another Point (e.g., Origin or a Corner):**
  If a shape is rotated around an arbitrary point, it may shift in space, leading to **undesired displacement** instead of pure rotation.

Role of the Centroid in Scaling

- **Scaling About the Centroid:**
  When a shape or group is scaled around its centroid, all points move outward (expansion) or inward (contraction) relative to the centroid, ensuring uniform scaling. The structure remains **centered** and balanced.
- **Scaling About Another Point:**
  If scaling is performed around an arbitrary point (such as the origin), the entire shape may **shift** unpredictably, leading to asymmetric distortion.

Groups

When dealing with a group of primitives, transformations should **preserve the relative positioning** of the shapes.

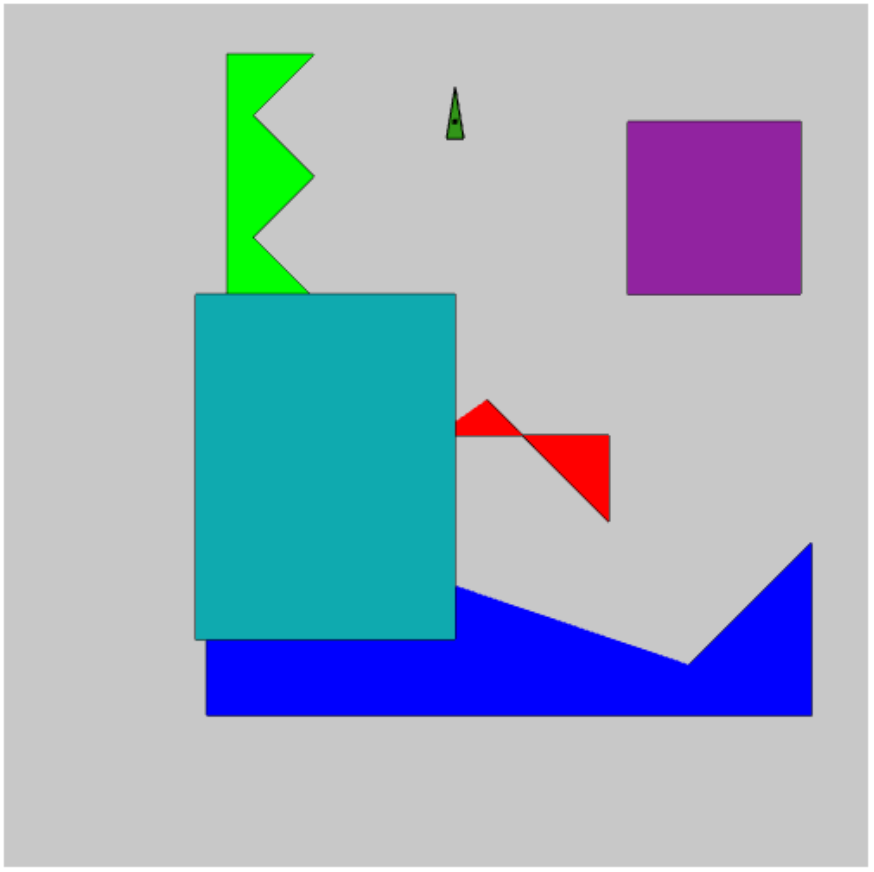
- Using the **centroid of the entire group** ensures that transformations apply uniformly across all shapes.
- Applying transformations relative to an external point (e.g., the world coordinate origin) may break internal alignment, causing the group to stretch or rotate in a disjointed manner.

References

- Tutorial Sessions Template
- https://sourceforge.net/p/jsclipper/wiki/documentation/
- https://github.com/mapbox/earcut
- https://youtube.com/playlist?list=PLjcVFFANLS5zH_PeKC6I8p0Pt1hzph_rt&si=qr92M5zjNi28IZoj
- https://webglfundamentals.org/

Screenshots and Video

https://youtu.be/9d81x2SMQFI



(Check instructions at the bottom of the page.)

Mode [m]: Draw Max Shapes: 6 Update

Drawing Tools

Brush On/Off [b] Brush Visibility [v] Fill Color

Turn Angle [← →]: 45 (°) Turn Reset Displacement [↑ ↓]: 50 (px) Move [w]

Shape Manipulation

Shape index [Num Keys]: 1

Angle: 45 (°) Rotate [→] Translation (px): 50 Translate [w]

Scale factor: 2 Scale [s] To Front [f] To Back [b]

(Select Fill Color in Drawing Tools) Change Color [c] Remove Color [r]

Scene Manipulation

Rotate Scene [→] Translate Scene [w] Clear Scene [c]

- Turn off brush to fill tha new shape with fill color
- Keyboard buttons assigned are written like this [keyboard key]
- Mouse click selects object
- Drag mouse to draw (better with starting drag on cursor)