|  |  |
| --- | --- |
|  | ICS2311 – Computer Graphics |
|  |  |
|  | Akech Atem  SCT211-0535/2022  3/13/25 |

## **CAT on Transformations and Shapes**

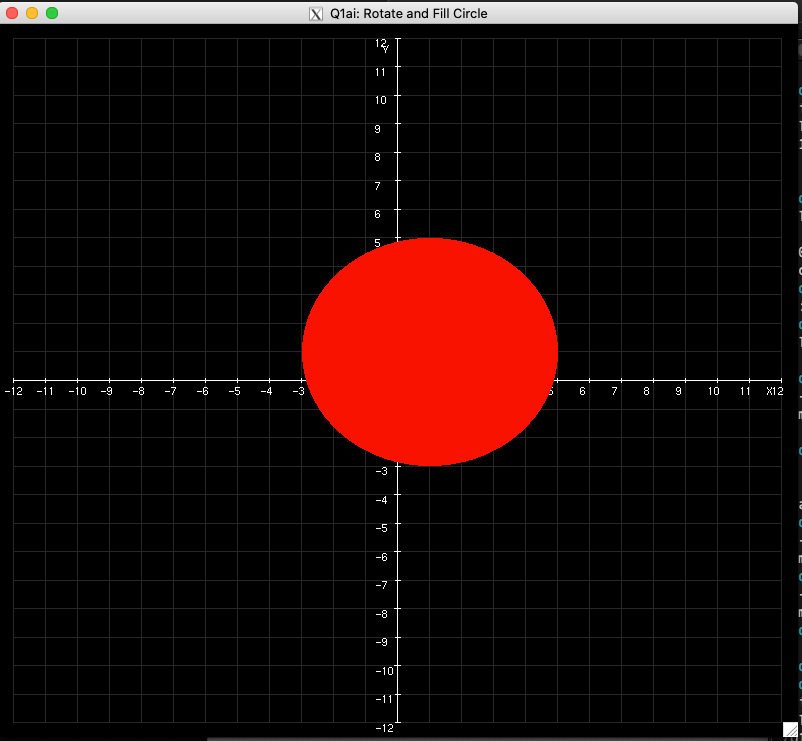
### **1: Bresenham Circle**

### **1(a)(i): Rotate and Fill the Circle**

#### Approach & Implementation

* Rotated the **circle 60° clockwise** using OpenGL’s glRotatef function.
* Filled the circle with **red color (#FF0000)** using GL\_TRIANGLE\_FAN.
* **Rotation is centered at (1,1)** for accuracy

#### Output & Screenshots

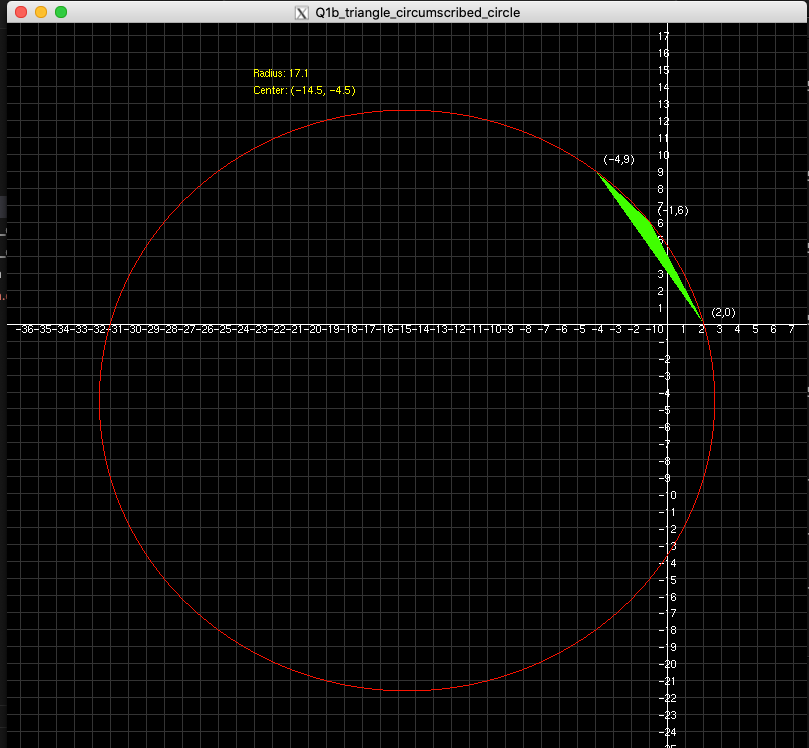


### **1(b): Triangle with Circumscribed Circle**

#### Approach & Implementation

* **Define the Triangle** using given vertices (-1,6), (2,0), (-4,9).
* **Calculate the Circumcenter and Circumradius** using perpendicular bisectors.
* **Draw the Triangle** with GL\_TRIANGLES.
* **Draw the Circumscribed Circle** using GL\_LINE\_LOOP and parametric equations.
* **Render the Cartesian Plane** with labeled axes for reference.
* **Label the Circumcenter** with computed coordinates.

#### Output & Screenshots

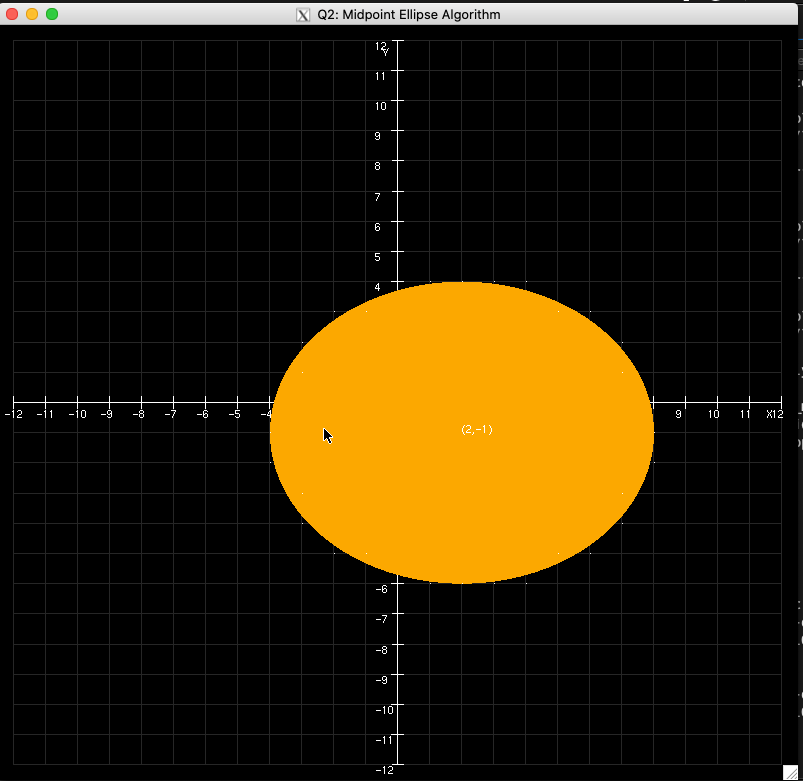


### **Question 2: Ellipse Drawing**

#### Approach & Implementation

* **Implement the Midpoint Ellipse Algorithm** to plot the given ellipse.
* **Apply transformations** to shift the center of the ellipse to **(2, -1).**
* **Ensure proper scaling** so that the ellipse fits within the Cartesian plane.
* **Fill the ellipse with an orange color** (#FFA500).
* **Label the ellipse center and axes** for clarity.

#### Output & Screenshots



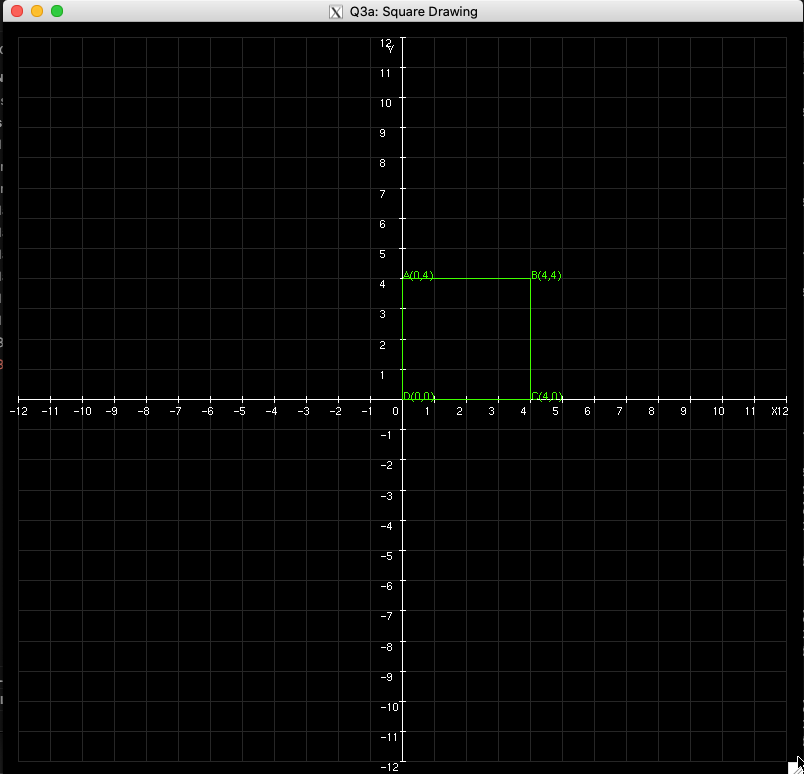
### **Question 3: Square Transformations**

### **3(a): Square Drawing**

#### Approach & Implementation

* **Define the Square** using given vertices **A(0,4), B(4,4), C(4,0), D(0,0)**.
* **Draw the Square** using GL\_LINE\_LOOP to outline it without filling.
* **Integrate the Square into the Cartesian Plane** for proper visualization.
* **Ensure the Cartesian Plane** has grid lines, axes, and small perpendicular ticks.
* **Label the Square’s Vertices.**

#### Output & Screenshots



## **3(a)(i): Translation**

#### Approach & Implementation

* **Apply Translation Transformation** using:

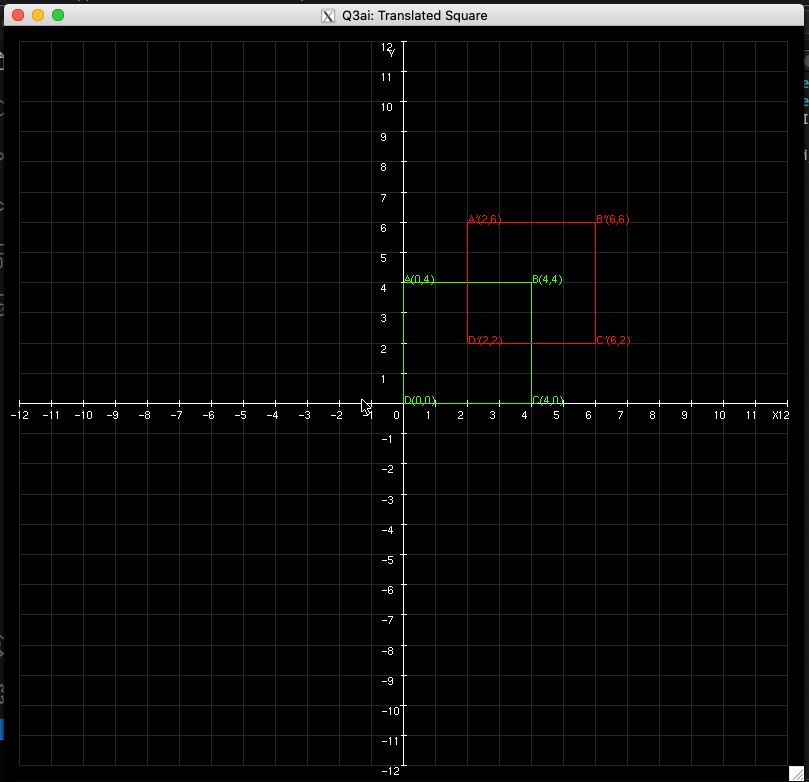
𝑥′ = 𝑥 + 2, 𝑦′ = 𝑦 + 2

* **Compute New Coordinates:**

𝐴′(2,6), 𝐵′(6,6), 𝐶′(6,2), 𝐷′(2,2)

* **Draw the Translated Square** in a different color for distinction.
* **Label the New Translated Vertices**

#### Output & Screenshots

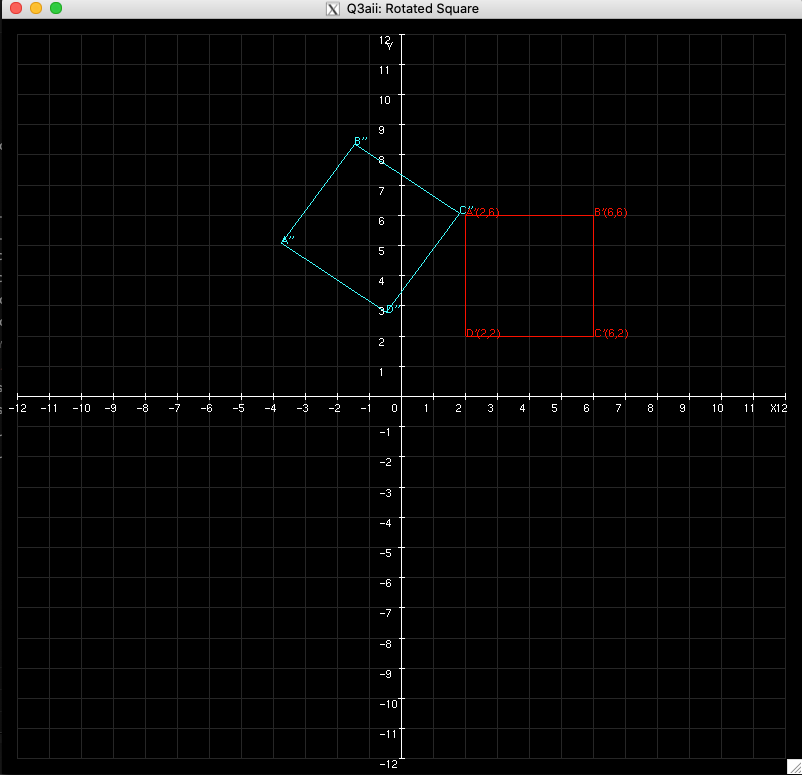


### **3(a)(ii): Rotation**

#### Approach & Implementation

* **Apply Rotation Transformation** using:  
   𝑥′ = 𝑥 cos(55) − 𝑦 sin(55), 𝑦′ = 𝑥 sin(55) + 𝑦 cos(55)
* **Compute New Rotated Coordinates:**  
   𝐴′(𝑥′,𝑦′), 𝐵′(𝑥′,𝑦′), 𝐶′(𝑥′,𝑦′), 𝐷′(𝑥′,𝑦′)
* **Draw the Rotated Square** in a different color for distinction.
* **Label the New Rotated Vertices** for clarity.

#### Output & Screenshots



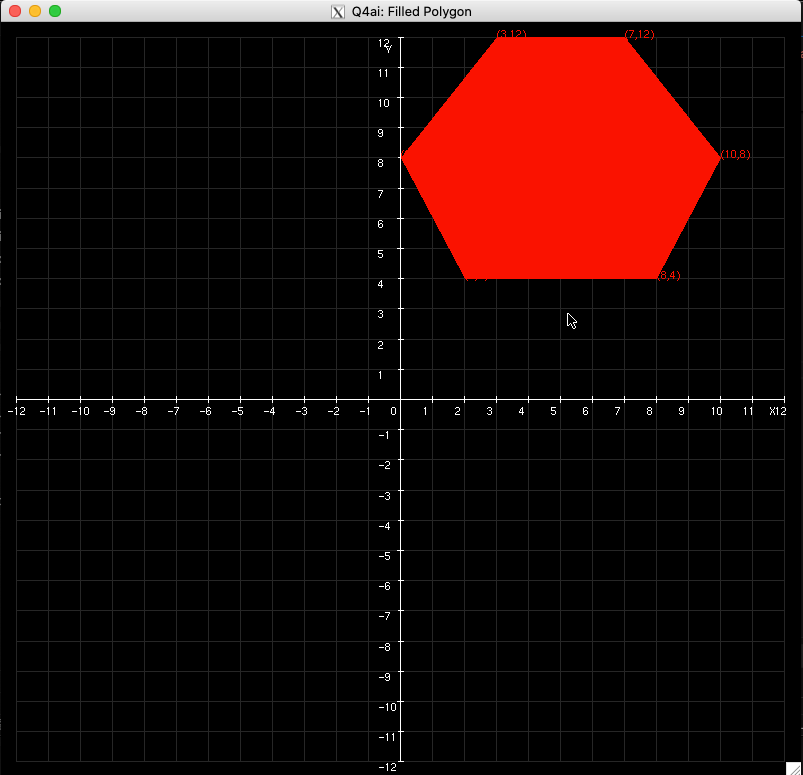
### **Question 4: Polygon Drawing and Scaling**

### **4(a)(i): Polygon Drawing**

#### Approach & Implementation

* **Define the Polygon** using the given vertices **(8,4), (2,4), (0,8), (3,12), (7,12), (10,8)**.
* **Use GL\_POLYGON** to draw the filled shape.
* **Integrate the Polygon into the Cartesian Plane** for proper visualization.
* **Fill the Polygon with Red (#FF0000)**.
* **Label the Polygon’s Vertices**

#### Output & Screenshots

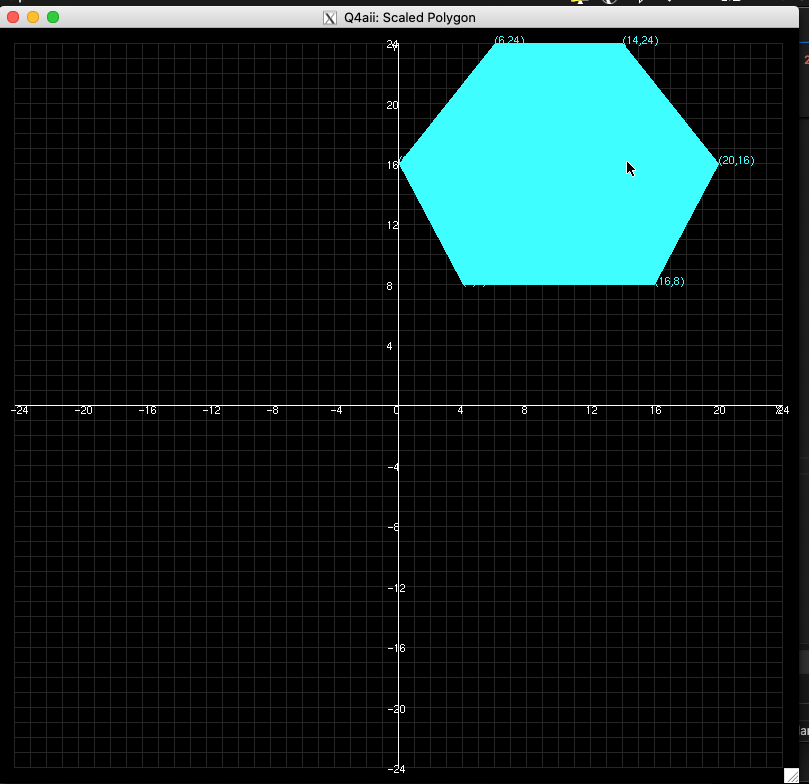


### **4(a)(ii): Scaling the Polygon by a Factor of 2**

#### Approach & Implementation

* **Apply Scaling Transformation** using:  
   𝑥′ = 𝑥 × 2, 𝑦′ = 𝑦 × 2
* **Compute New Scaled Coordinates:**  
   𝐴′(16,8), 𝐵′(4,8), 𝐶′(0,16), 𝐷′(6,24), 𝐸′(14,24), 𝐹′(20,16)
* **Draw the Scaled Polygon** in a different color for distinction.
* **Label the New Scaled Vertices** for clarity.

#### Output & Screenshots

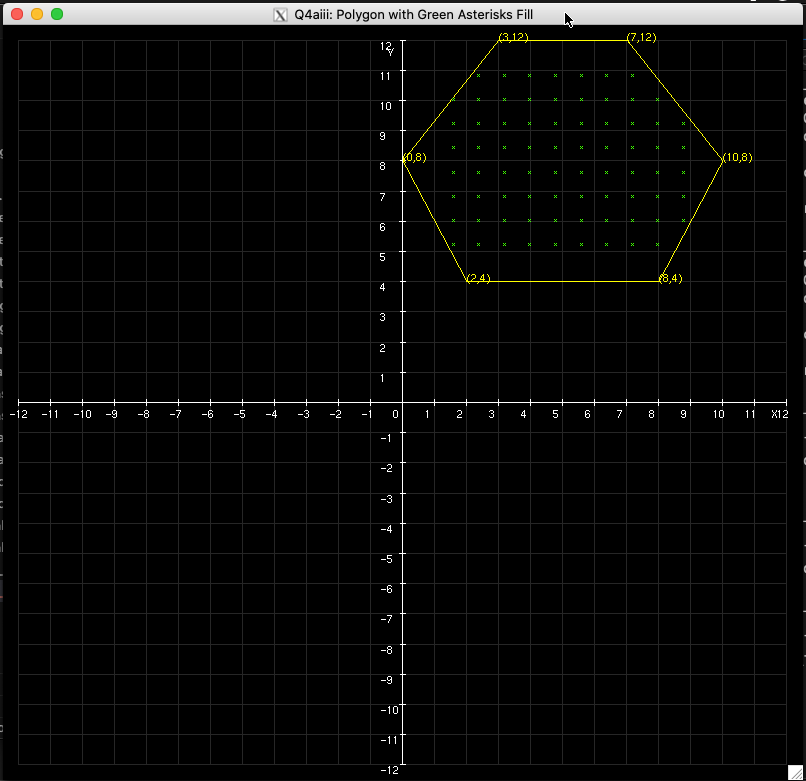


### **4(a)(iii): Filling the Polygon with Green Asterisks**

#### Approach & Implementation

* **Use Raster Positioning (glRasterPos2f)** to place asterisks inside the polygon.
* **Iterate through a grid within the polygon’s bounding box** to determine fill points.
* **Use glutBitmapCharacter to render ‘\*’ at valid positions.**
* **Maintain the Cartesian Plane for reference.**
* **Label the Polygon’s Vertices** for clarity.

#### Output & Screenshots



### **Question 5: Parabola Drawing**

#### Approach & Implementation

* **Implement the Midpoint Parabola Algorithm** to plot the curve ( x = y^2 ).
* **Use symmetry** to reflect points across the x-axis for completeness.
* **Integrate the parabola** into the Cartesian Plane for reference.
* **Ensure proper scaling** so the parabola is well-positioned and visible
* **Label key points** on the curve for clarity.

#### Output & Screenshots

