**Approach Method**

1. **Data Preparation:**
   * **Input:** Read polyline data from CSV files.
   * **Output:** Convert polyline data into cubic Bézier curves and SVG format.
2. **Curve Regularization:**
   * **Identify Regular Shapes:**
     + **Straight Lines:** Use linear regression to detect if points form a straight line.
     + **Circles and Ellipses:** Fit points to circle/ellipse equations using least squares fitting.
     + **Rectangles and Rounded Rectangles:** Identify sets of parallel and perpendicular lines, and check for rounded corners.
     + **Regular Polygons:** Check for equal-length sides and equal angles.
     + **Star Shapes:** Identify central points with radial symmetry and multiple arms.
3. **Symmetry Detection:**
   * **Reflection Symmetry:** Find and verify axes of symmetry by mirroring shapes along potential symmetry lines and checking for overlap.
4. **Curve Completion:**
   * **Detect Incomplete Curves:** Identify gaps or occlusions in the shapes.
   * **Complete Curves:** Use curve fitting and interpolation techniques to fill in the gaps smoothly and naturally.
5. **Visualization and Output:**
   * **Visualization:** Plot the identified and regularized shapes using matplotlib for verification.
   * **SVG and PNG Conversion:** Convert the regularized and completed curves to SVG and PNG formats for final output.