

# *Computer Graphics*

---

## (Graphics Primitives – Filled Areas) Lecture 003

# *Filling Polygons (Odd-Parity Rule)*

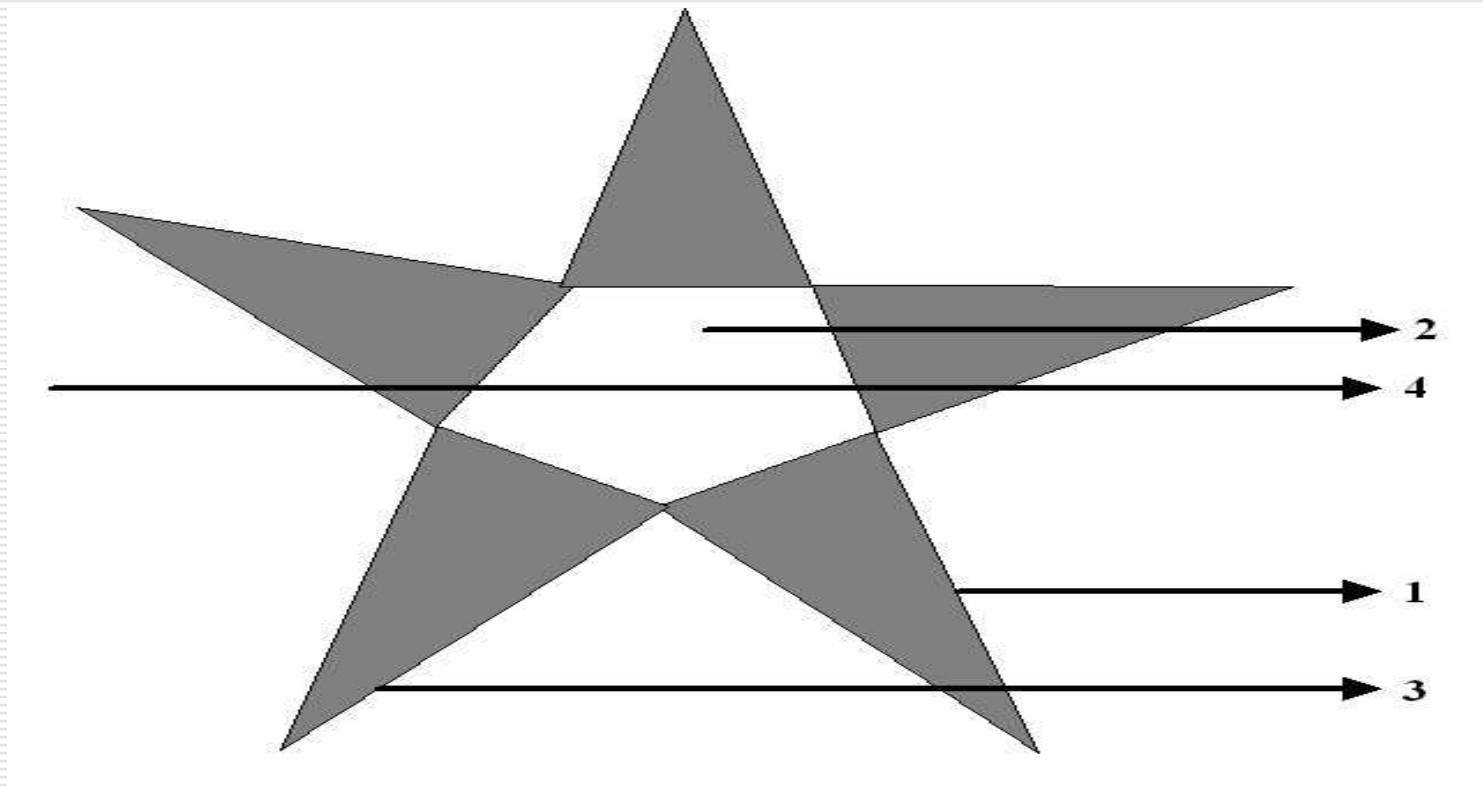
---

- ❑ The algorithm needs to decide whether a particular point is interior or exterior to a polygon. We need to determine which pixels belong to which primitive (polygon)
- ❑ An easy algorithm to determine whether a particular region is within a polygon is that adopting the inside-outside test. A rule called the odd-parity (odd-even rule) is usually applied to test whether a region (point) is interior or not.
- ❑ With this test, a point is considered to be interior if the number of intersections between the line and the polygon edges is odd. Note that the line **MUST NOT** intersect and polygon vertices.

## *Filling Polygons (Odd-Parity Rule Example)*

---

- The first two points are outside while the 3<sup>rd</sup> and 4<sup>th</sup> points are inside polygons.



# *Scan-Line Polygon Filling Algorithm*

---

- Horizontal scanning of the polygon to be filled from its lowermost to its topmost vertex.
  
- For each horizontal scan line:
  - 1. List all the points that intersect with the horizontal scan-line.
  - 2. Sort the intersection points in ascending order of the x coordinate.
  - 3. Fill in the interior pixels between pairs of successive intersections.

The third step accepts a sorted list of points and connects them according to the odd-parity rule.

## *Scan-Line Polygon Filling Algorithm Example*

---

- ❑ Step 1 can be optimised by considering a list of sorted edges instead of points intersecting the scan-line.
- ❑ Consider example polygon on page 24 of notes.
- ❑ Note that E8 and E4 are not taken in consideration when filling.