Lect 2-3: Images
Lect 4-5: Basic shapes
Lect 6-9: Camera projection
Lect 10-15: Illumination
Lect 16: User interfaces
Lect 17-21: Curves, surfaces, fractals, and dynamic objects
6. Scan-converting polygons (or Polygon filling)

1. Basic Idea: Intersect the scan-line with each polygon edge, and draw pixels between intersections.
   1. Apply “on-off” walk: The pixels that are inside the polygon are between each odd-even intersection pair

2. Algorithm sketch
   1. Find out min, max y coordinate of the polygon
   2. Increment y from ymin to ymax
      1. Update the list of edges that intersect with the scan line
      2. Use line-scan-conversion to draw pixels on each edge with that y-coordinate
      3. Sort the edges by the x coordinates of the first pixel on the scanline.
      4. Draw pixels between the pixels of odd-even edges
3. Problems and solutions
   1. What happens when scan-line passes through a polygon vertex, how many intersections should be counted?
      1. For each edge, only count the vertex with larger y coordinate than the other vertex.
      2. Ignore both vertices of a horizontal line.

   No intersections

   1 intersection

   2 intersections