Claudia & Jack

Things to address:

* Matching template tiles to target image
  + Need a way to resize template to fit the tiles of the target image
    - Account for the width
  + Do template matching algorithm from class
* Finding width
  + Take gradient of a sub-section of image (make assumption that the whole strand width is constant
  + Find point w/ non-zero gradient
  + Move in direction of gradient until you hit a point w/ non-zero gradient
  + Repeat if non-zero gradient isn’t opposite of direction of gradient
* Finding distance between two strands

Tile representation:

* Have data w/ each table of splines w/i tiles
  + Splines are located on the edges and corners of tiles
* Nodes are corners
* Points are center of tiles (traverse each point in the list)
* Edges are linked lists of corners

Steps

Part 1

* Width and tile size detection - Jack
  + Width of strand
  + Width of space between strands
  + Width of space at overlapping points
* Make the tile template skeletons (nodes and edges) - Claudia
  + Make the template images based on the template
  + How to link the template image to the skeleton?
  + For each skeleton
    - Make the template
    - Template match
    - Find locations where it fits
  + Somehow map the templates in a grid (centered?)
* May need to make adjustments to the original image to get the borders to auto fit (space the outer edges to be strand width / 2 + the spacing between strand
* Matching the tiles to the target image

Part 2

* Apply algorithm to determine the edges and nodes
* Find separate strands
* Changing the width of the strands