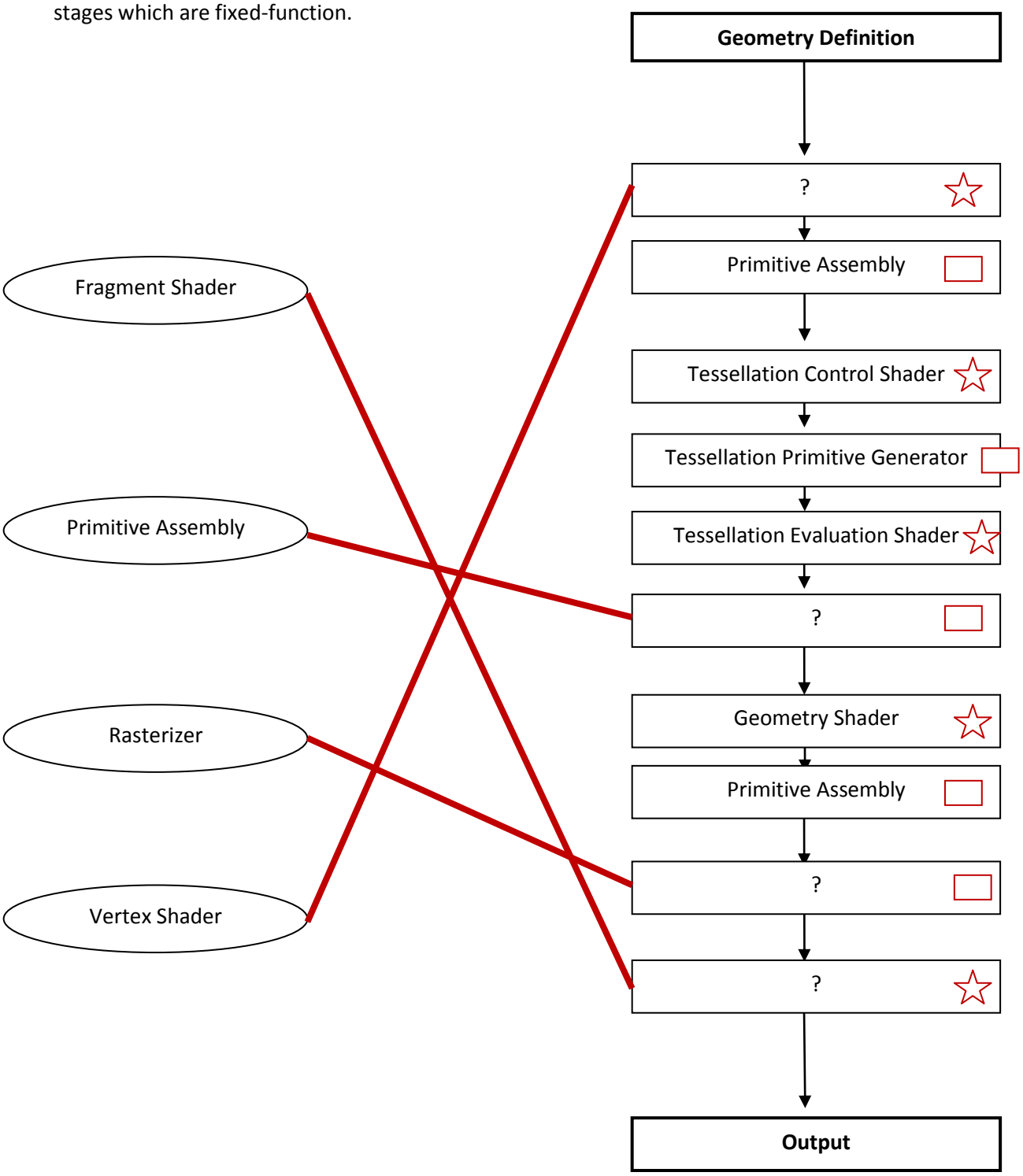
Names: Answer McSheet, Dani Witherspoon

Lab 8 Shaders Workshop

**WORK IN TEAMS OF TWO.**

**1a**. Draw a **line** from the elements to their proper place in the OpenGL pipeline.

**b.** Draw a **star** next to the stages of the pipeline which are *programmable*. Draw a **square** next to those stages which are fixed-function.



**2.** Annotate the following functions below according to where they are typically done:

| | |
|---|---|
| V | Vertex transformations |
| FP | Viewport mapping |
| FP | Depth test |
| V | Normal transformation |
| F | Color computation |
| FP | Backface culling |
| FP | View volume culling |
| F | Texturing |
| V | Per-vertex lighting |
| F | Per-pixel lighting |
| V | Normal normalizations |
| F | Discarding pixels in fragments |
| V | Texture coordinates |

**V** *if it is typically done in the vertex shader*

**F** *if it is typically done in the fragment shader*

**FP** *if it is still done by the fixed pipeline*

**3a.** In one sentence, explain "in" and "out" variables in GLSL.

In and out variables are how shaders receive and pass the variables they will edit / pass on to the next shader in the pipeline.

**b.** In one sentence, explain what a "uniform" variable is in GLSL.

A uniform variable is a variable which will be used by at least one shader (but possibly many), which is sent from the application code to the shaders.

**4.** Fill in this Venn Diagram with **at least** six statements about the language features of C/C++ and GLSL: