You have 70 minutes. No computers, notes, or other aids are allowed.

Except where otherwise noted, any question about our software graphics engine refers to the most recent version. Because this exam covers all course material up to and including depth buffering, the "most recent version" is the one used in 130mainDepth.c — incorporating depth buffering but no later features.

The best answers are not just factually and logically correct. They emphasize the most important information, de-emphasize less-important information, and exclude irrelevant information. They convince the grader that the student understands the context and motivation of the question. They are concise. Frequently they employ diagrams or pictures.

If you believe that a question is ambiguous, then ask for clarification. If the clarification does not help, then explain your interpretation of the problem in your solution.

Good luck. :)

A. Draw a dependency graph of our software graphics engine. The nodes are the files in our engine. Add an arrow from file A to file B if A depends on B (that is, A uses constants, data types, or functions defined in B). For clarity, try to arrange your graph so that most of the arrows point down the page

B. What problem does depth buffering solve? What other solutions have we discussed? Why do we prefer depth buffering over them? Remember that pictures can help explain.

C. The fragment shader has access to uniforms, varyings, and textures. Why haven't we given the vertex shader access to textures? Explain.

D. I can think of two distinct algorithms for implementing `meshRender`. Each algorithm has advantages and disadvantages relative to the other. Explain.

E.A. In our graphics engine, are there any restrictions on how attribute vectors are formatted? I mean, by editing main.c but no other files, can you build a mesh with any attributes you want, in any order, write the vertex and fragment shaders accordingly, and end up with a working program?

E.B. Similarly, are there any restrictions on how varying vectors are formatted?