Conceptual Questions:

1. What visual errors may occur when using linear blend skinning for a mesh? Why do these errors occur?

    LERPing a rotation matrix causes scale skewing to occur, this will cause the mesh will collapse as joints are rotated.

    Because when a joint is rotated, the vertex connections will become crossed and it will not preserve the volume as usual.

2. Since one cannot insert breakpoints into a GLSL shader program, how might one debug a shader? For example, if one were writing a vertex shader that applies a mesh skinning deformation, how might one determine which vertices are influenced by a particular joint? Consider what alterations to your fragment shader might be useful to test this.

    We can debug the shader program in mygl.cpp, since we created the prog_flat or prog_skeleton to compile the shader, we can set a break point after we set the prog_flat, for example, we can check the attributes of it to check what is going on on the shader program. Besides, to check which vertices are influenced by a particular joint, we can figure this out before we send the vbo of influencers and weights to the shader, it is in the mesh.cpp, the create() function.

Code Requirements:

(there may be some problem to run the program, but it works very well in Debug, I don't know why.)

    4.1. Done.

    4.2. Done. The Root Joint is not a real Joint, I just set it as the Root, its first child should be the first Joint of the skeleton, the hip.

    4.3. Done.

    4.4. Done. When you click Skinning, the weights and the influencers will be set correctly, because the bug of the 4.6, you can only see the results in Debug and also I output result in windows.

    4.5. Done. There are 6 button to move the joint.

    4.6. Done, but there is a very strange problem. The prog_flat.draw will be different when I change the prog_skeleton. I did write all the code correctly, including the shaderprogram.cpp, the create() in mesh.cpp, and the prog_skeleton's settings in mygl.cpp, the binding the passing of the data should all be correct, too. This was checked by one of the TA, Trung and the Professor, Adam. But the result was very strange, I worked on this for more than 7 hours ,the TA for 2 hours and even Adam for about 1 hour, and nobody knew the reasons, because the code seems nothing wrong. If you are interested about this problem, you can check my code or ask Trung about it.

Extra Credit:

Because the big bug of the 4.6, I didn't want to spend more time on these three questions.