

CS250 A2

Worth: 10 points
Due Date: 02/27/2006
Due Time: 12 noon
Hardcopy: Place your hardcopy version in the appropriate CS250 submission box located in the front office. Don't forget to sign the declaration on page 4 of this document.
Softcopy: Copy your zipped softcopy version to your CS250A/B network folder.

Topics

- Introduction to Graphics Pipe
- Transforms: Scaling, Rotation, and Translation
- View Transform
- Perspective Transform
- Viewport Transform
- First Person Camera
- Triangle Rasterization and Color Interpolation
- Hidden Surface Removal using Back-Face Removal & Depth Buffer

CS250_A2 Framework

- This framework has been designed and implemented to provide a code base for CS250 students in implementing their assignments.
- It implements a complete software-based graphics pipe. For each assignment, certain features previously implemented by us will be removed and you will be assigned the task of implementing these features. The software renderer is included as a library file in the project: **GfxLib.lib**. It will be linked to your contribution to create an executable.
- Please note that this framework has been designed with the single intent of providing an educational tool. It is not suitable for use as a high-performance software renderer for applications such as games.
- To report bugs and for answers to any questions you might have about the framework, contact either myself or the CS250 TA: Patrick Laukaitis (plaukait@digipen.edu)
- ***It is required that all CS250 students use this framework for implementing, testing, and submitting CS250 A2.***
- The framework uses the right-handed coordinate frame explained in class: X-axis is oriented to the right; Y-axis is oriented into the paper; Z-axis is oriented vertically up.
- For A2, implement a first person camera using spherical coordinates (as explained in class) and triangle rasterizer with back-face removal, hidden surface removal (depth buffer) & color interpolation. Depth and color interpolations must be implement using linear (not, bi-linear or nested) interpolation.
- For testing purposes, an executable file, **CS250_A2 Sample.exe**, is included

containing our correct implementations of a first person camera and a triangle rasterizer with color interpolation.

- **CS250_A2 Sample.exe** uses the following controls:
 - 1 through 5: Switch between displayed models.
 - Mouse: Orient camera using spherical coordinates.
 - Mouse Left & Right Buttons: Displace camera along and opposite direction of view vector, respectively.

Statement

Do the following:

- First, open the Visual Studio Solution Object containing this assignment's project: **CS250_A2 Framework.sln**. The project includes all required libraries. Compile, link, and execute the project. This executable should be similar to the included sample executable. To incorporate your implementation into the project, complete the following steps.
- **First Person Camera**
 - Read declaration of base camera class, **gfxCamera**, in **.\GfxLib\gfx\Camera.h**
 - Read declaration of derived camera class, **YourCamera**, in **.\GfxDriver\src\Camera.h**
 - Implement the derived camera class, **YourCamera**, in **.\GfxDriver\src\Camera.cpp**
 - Call your camera implementation by un-commenting lines 224 and 228 in **.\GfxDriver\src\main.cpp**
- **Triangle Rasterizer**
 - Read declaration of vertex class, **gfxVertex**, in **.\GfxLib\gfx\Vertex.h**
 - Read declaration of base rasterization class, **gfxController_Rasterization**, in **.\GfxLib\gfx\Controller_Rasterization.h**
 - Read declaration of derived rasterization class, **YourRasterizer**, in **.\GfxDriver\src\Rasterizer.h**
 - Implement the derived rasterization class, **YourRasterizer**, in **.\GfxDriver\src\Rasterizer.cpp**
- **Miscellaneous Notes**
 - Search for "**@todo**" within source files in **.\GfxDriver\src*.cpp** to identify those location where you are expected to insert your own code.
 - There are two locations (lines 224 and 228) in **.\GfxDriver\src\main.cpp** where you must uncomment lines to enable your own implementation (as opposed to the reference implementation). **DO NOT FORGET TO DO THIS.**
 - Be aware that you must implement the viewing/projection/ndc matrix functions of your camera prior to using your camera implementation (otherwise you won't get anything on the screen).
- You can compare your implementations with the correct versions by running **CS250_A2 Sample.exe**.
- Provide a **manual** page with your hardcopy submission. In your own words,

describe what has been implemented, how it was implemented, any modifications made to the assignment specifications, cool features implemented, etc. In addition, the manual must clearly specify any portion of the assignment that was not completed/attempted and/or incomplete. Don't forget to include derivations with proper pictures for derivations of view, perspective, and viewport transforms, conversion from spherical to cartesian coordinates, linear (not, bi-linear) attribute interpolation, and back-face removal. Note that if we can't read what you write, we can't grade your assignment.

Deliverables:

- Softcopy: Zipped folder containing the entire CS250_A2 Framework with object code and executables deleted. We are going to unzip your folder, double-click on **CS250_A2 Framework.snl** to open the project and then compile, link, and execute your assignment.
- Hardcopy: Stapled documentation containing all relevant pages as specified in the assignment submission guidelines.

Objectives: CS250 A2

Name: _____

Login Name: _____

SRS Student ID: _____

___1: Correct information was handed in as hardcopy and a README file exists in student directory. If not, penalty is **–20 points**.

___2: Student program compiles, links and executes. If not, penalty is **–100 points**.

___3: **First Person Camera is implemented correctly. Worth 50 points.**

___4: **Triangle Rasterization with Hidden Surface Removal, Back-Face Removal & Vertex Color Interpolation is implemented correctly using linear (not, bi-linear) interpolation. Worth 50 points.**

___5: A good manual page was submitted. If not, penalty is **–30 points**.

DECLARATION:

I have read the statements regarding cheating in both the CS250 course handout and DigiPen student handbook. I affirm with my signature that this is my own solution to A2 and the submitted source code is of my creation and represents my own work.

Signature: