Scott Bebington u21546216

Class Test 3

Camera View

Third Person Viewing: This viewing angle provides a good perspective on how the rover interacts with the surfaces and the user is able to see everything happening on the rover with a small view of what's behind the rover.

First Person Viewing: This viewing angle provides a smaller viewing angle, but it puts the user in the shoes of the rover, so they are able to see everything the rover sees, making it more realistic.

Top-Down Viewing: This viewing angle is great to see the surrounding area of the rover however this may cause the objects as well as the rover to look 2D and it removes any depth from the program.

My choice would be a first-person viewing approach with the ability to swap camera angles from the main viewing camera to the blind spot cameras. This means the users gets an immersive experience and it keeps the rendering costs relatively low as multiple views don't have to be shown at once.

Because of this choice, when moving the rover. Very few calculations have to be made as instead of moving the rover to the origin and performing calculations there, it is already centred. This means all that needs to be done is to translate the surrounding and calculate the rover's y value if the height of the terrain changes.

Robotic Arm Rending and Animation

Single structed arm: For this approach there would be one long structure for the arm and the only movement would be at the joint where the arm of the rover meets the body, this approach involved very little movement for the arm but would lower the render cost.

Multi structured arm: For this approach instead of having 1 long structure for the arm, split it up to make it more similar to the human arm in which the upper arm can move up and down and the forearm can move side to side, this would increase the movement of the arm creating a more realistic movement, but it would also increase rendering costs.

For the sake of realism my choice would be the multi structured approach. It created a more realistic feeling which is beneficial in modern times.

Tools:

The approach I would take for rending the tools on each arm would be to use a revolver type approach, so there would be 6 tools on each arm for instance and the user would be able to rotate through each tool until the one they want is loaded. Then they would perform the action used for that tool.

Snake and insect shape rendering and animation.

Cylindrical shapes: This option for rendering the snakes and insects would make them look more realistic, however, the rendering costs would be relatively high as cylinders are high polygon shapes, texture mapping would also be more difficult as any textures would have to be wrapped around any circular shapes, following this the animation of legs moving on an insect and the body of the snake moving would make the calculations for movement and textures very complex.

Triangle strip: using a triangle strip is a much more simplistic approach, it keeps the rendering costs lower as no arc calculations need to be made, the texture mapping becomes easier as it being used on a flat surface, the only downside is the need for shearing when the insect or snake moves, causing some calculations to be made.

Overall, the Triangle strip approach is the one that I would take. It is simpler to implement and overall would cause less lag due to it not having as many calculations.

Snake and insect textures and details.

I would use texture mapping to achieve a realistic look for the insects and snakes, by taking images of snakes and insects and mapping the images to coordinates of the body will cause the snakes and insects to look very realistic, this eliminates the need for colouring each section separately and worrying about the look of the scales or skin.

A similar thing would happen with insects. The only difference being if the creature has wings the texture and the overall shape would have to somewhat transparent in the case of clear wings.

Landscape shape rendering.

Primitive shapes: This involves using something like triangles strips to create a sort of map that, this is good as there are no gaps in the terrain and it make it easier for texture mapping as there is no warping and no complex calculations need to be made, this is also a low render cost as no 3D shapes need to be created, only many 2D triangles.

Landscape textures and details.

Using coordinates for texture mapping: If I'm working with the basic shapes mentioned earlier, I'll apply textures by directly mapping them using coordinates. This approach simplifies the process, as there's no need for extra calculations when rotating or scaling the textures, and there won't be any shearing effects either.

Culling

Because of the design choices I've made earlier there isn't much culling that needs to happen. The only culling that would need to occur is for the object not in the main field of view would not need to be rendered until they come into view, this is also the case for the inserts or snakes, if they are not within the viewing angle they do not need to be rendered. When switching views the objects within the blind spot cameras would need to be in view and the objects within the main view would need

to be culled. With regards to the terrain, because the terrain is relatively simple and has low rendering costs, there isn’t any need for it to be culled.