Michael Maris
Programmed and refined the code for essential gameplay mechanics, including:

- Grabbag.py: Implemented the logic for randomizing set pieces.
- ActionCommand class: Implemented the logic for storage, testing, and retrieval of quick-time events, as well as linked it to the gameplay.
- GameHUD.py: Implimented code for the GUI, including the function that updates the GUI.
- Game.py: Implimented collision detection, scoring, and the in-game timer that tracks how long quick-time events last, as well as the meathod of passing these values into the GUI.
- Title Screen: Implemented the code for displaying the title screen, and also contributed to the design of the title screen.
- PipeGeneric.py/Game.py: Streamlined the meathod by which individual pipes can be identified.

Harrison Krug
2D art assets, hud design, and sound effects.

- Created the title screen.
- Assisted in the design of the HUD and created art assets for the stability bar and quicktime events, as well as helping write the code to blit them on the screen.
- Acquired all sounds effects with the exception of background music.

Sebastian Basch
Particle Effects

- Created,implemented,and lit all particle systems
- recolored the stability bar
- created the prototype hud with buttons instead of bars, as well as translucent side pieces (not used in the final version)

Tif Entwistle
Art Forever

-Modelled, UV'd, rigged, skinned, textured, sculpted, spec'd, animated the main mechanical spider character
-Modelled, UV'd, textured, bump'd, spec'd, self-illuminated/incadescence'd the modular environment tunnel & broken bits
//it's cool, because the entire environment shares one texture/spec/bump/incandescence
-Created Game over screen without text/etc.
-Created Infinity Plane

-Concept art'd the entire thing, though there aren't pics of that in the file.
-fought with Maya and Panda3D tooth and nail to get what we have

Kevin Hendricks
Main Game Code, Team Leader

Everything else on the project, including the Pipe Classes, almost all of the lighting, all code concerning the spider and the majority of interactions between objects, implemented sounds and animations, and general implementation of all the separate elements others worked on and general sprucing.
All the main algorithms to determine what gets produced where in what order and orientation to make a believable infinite tunnel with a non-bias distribution of elements and recycling as many elements as possible, bringing the frame rate up from unplayable when we first introduced particle systems to only occasional minor bumps.
Organized all the team meetings and fielded many questions from others. View the repository commit history for more detailed information.