CSSE1001: Assignment 3 – Reflections

17/10/08

Name: Jonathan Keller

Student Number: 40798545

Project Title: CAM software…
AutoCUT 1.0: An Automated Bandsaw Software Package

Reflections

Overall I was very happy with the end result of what I achieved in this project throughout the semester. I put a lot of work into my program and even more thought and, as it was the first genuinely functional program that I had written, it was incredibly rewarding to see it all come together with a shape originally created in a commercial program, being “cut” out by a virtual saw blade, smoothly, at warp speed.

My skills as a programmer have increased dramatically throughout the course of the project and I have no doubt that if I started from scratch today, I would implement something (more easily) that was better, with in particular, better structure. When I started the project I didn’t really understand class structure or GUI design at all, and I think this is reflected in the layout of the mainWindow and virtualCut classes, and has unfortunately propagated through to the smaller dialog box classes, in the form of some small issues. When I have the dialog box classes inherit from Toplevel they cannot initialise Toplevel with the mainWindow as a master. I couldn’t get the main window to inherit from Toplevel without ruining the splash screen function so I decided to leave it as is, so the main window pops up a little bit smaller than the screen, which isn’t too bad, but still something I want to neaten up eventually.

Even though some program functions are not particularly efficient, everything runs at an acceptable speed, the only thing that is noticeably slow is the animation loop towards the end of a large shape, because the white cut-trail is a list of coordinates that grows by 2 every time the animation loops. This is improved temporarily by increasing the node resolution a little, and could be improved by having an inspection function, that checks if any two points are in a straight line and if they are, removes the coordinates of the points in between. I implemented most of the calculation functions outside of the class structures as stand-alone functions, as I found that this improves speed of functions that are called many times.

The bugs that remain in AutoCUT 1.0 are cosmetic ones. Lots of little things could be done to have a more professional interface, like: better looking user interfaces, a more robust system of disabled/enabled menu buttons, a picture in the about box, focus on new windows, full screen windows and centred and modal dialog boxes.

Before AutoCUT is expanded in future installations, I think the main window class and the virtual cut classes should be split into 2 separate files as they will most likely grow in size and it is already quite difficult to find exactly what you are looking for in the file of more than 1000 lines.

Overall I think I have created a solid basis for a very functional software package. Lots of additions are planned for the future:

- Full Screen defaulting windows
- Wood grain finish on the virtual cutting material
- Bandsaw noise during the cutting simulation
- Automatic filtering of draft lines (not hard-coded like it is ☹)
- Automatic filtering of lines that aren’t joined to other lines (including dimension lines)
- A manual system of clicking and removing lines from shape
- Support for other CAD interchange file types

- Support for b-splines
- Support for CAD files with more than one enclosed shape
- The option to click and select which enclosed shape is calculated
- Support for importing shapes from images
- Support for 3 axis bandsaw automation (x, y, rotation)
- Support for 3D CAD models
- More options, including automatic line filtering on/off
- A progress bar when calculating JCode
- Displaying loaded JCode files on the main canvas (not just imported IGES files)
- A "smarter" JCode calculation process which determines when a corner is to tight to cut with a bandsaw and backs up and cuts a temporary arc that doesn't go through the shape outline, to be able to actually cut shapes with a bandsaw
- A function that searches the trail list for any consecutive points that are in a straight line and deletes the middle coordinates
- And obviously integrating a mechanical bandsaw with the system to actually cut shapes

Conclusion

There were many times throughout my project that I wondered whether it would have been better to make a simple game or database or something altogether trivial, but the resulting program has been worth the trouble. The difference for me between writing AutoCUT and writing a trivial program is that an automated machine ties into my mechanical engineering studies and has helped me to put the software side of the product development cycle into a professional perspective.