Reflection Joshua Hicks 41749450

I’m actually really happy with my project. I haven’t done everything I wanted to, but I knew that I wouldn’t from the start. I’ve spent a lot more time on this than on most other subject this semester, but I also think I’ve learnt the most from it. Probably because I started fairly early in the term so I was forced to learn on a really sharp curve which let me understand the little tips and tricks presented in lectures.

The other thing I’ve learnt is how much better the code is every time you start from scratch because everything seems to fit so much nicer. That’s probably just getting more comfortable with python, but even going back last night and doing the comments, I could see a lot of places where things could have been more efficient. However, rewriting a project this big from scratch several times hasn’t always been fun. Every time I finally got to the end I realized how much better I could make it. Now I’ve heard from many teachers in the past that this is usually the result of a serious lack of planning, but I think that I just didn’t have the understanding and the only way I could begin to understand was just programming.

I guess that’s the only real way to learn programming and I’m glad for this project because it pushed me to program so much. My advice to all future students would be to start this one first. Especially when it seems impossible if they just keep googling and trying until they understand it. It probably saves you time in the long run, because after doing something like this you can laugh at the stuff in the first two assignments. And then do them so well that you learn more than you would have in the first place. Not to mention that at the end of semester other subjects have projects to be done…

Hmm, bugs in the program. There is one when you click on a menu, and still hold key and then you let go of the key and finally the quit the menu, but the cube keeps spinning. Twice the speed solver array has just stopped working. I don’t know why, and I’ve tried to figure out what went wrong, but I just can’t do it again. Something I would do differently if I were to do this again would be to make nearly everything run off events. I only really started reading about the custom events over the last little while and I think they would have made life much easier, not to mention the code more efficient. I think the bugs that might be hidden will be from events, so setting them up really well would be important.

One reason some parts of code was inefficient was because it ran faster written that way. For example, my Model class runs probably 2 to 3 times as fast as what it used to because of just writing out all the different face turns. The fact that it now uses pointers instead of making new copies every time makes memory not an issue no matter how long it is run. Of course figuring out how to use pointers with array was hard, probably the only reason I understand it in either 1000 or 1001 is because I did it in both. Some of the recursive code was less efficient than it could have been, but I find it hard to make good efficient recursive code. Also in this case, I never needed to fix the code after it had made all possible path files for that method of solving.

I’m a little disappointed about not having a better solve method for the cube. I did get one, but after programming it the whole way through it would have taken forever to actually run. However, at the moment, the program doesn’t really display its solver abilities, because it’s already managed to solve them all and now just does what it’s already done. It is much cooler to solve half a million cubes in 15 minutes compared to 1 cube in the same time, but I would have liked a more efficient solving method. However, apart from that I cannot really think of an area where this project isn’t complete. There are places where I could have added extra stuff, but it’s complete as it is.

I like the modularity of this project. However, if I were to do it again I’d probably have every cube visual include a cube model. I tried to do it separately because of the VMC concept that we learnt in assignment two, but I think they could be combined successfully. The problem was that everything was so interlinked, that to modularize it more would have made the program more complex than it needed to be. The only thing that really should be one object was the replay data holder and the record data holder, because one is derived from the other.

One thing I realize now that I’m finishing up is that there are a lot of modules out there that I could have used to make programming this so much easier. Some of the time to string functions using date time objects would have been easier. There are doubtlessly better 3D graphics modules than tkinter, even for simple objects like cubes.

The documentation, well, it wasn’t the most interesting part of this project. Of course, leaving it to the end wasn’t the smartest thing to do. I’m not really sure how to document programs like this though. I went through the code and commented everything and for the design document it seems I just listed out everything again. I was really surprised at how much code I actually had. And it was lots of the background stuff that I hadn’t worked on for a while, like the cube canvas. I can imagine how impossible it would be for someone to follow my coding without comments for that if they didn’t know what I was doing. But for some of the other stuff the code seemed self explanatory and I wondered how useful comments there actually were. I probably didn’t go into enough detail commenting some of the harder maths or Rubik’s concepts, and I didn’t think I was supposed to go through the whole programming commenting why I’d chosen to do something.

I’d like to say that my program is usable, but considering the number of people who can solve a physical Rubik’s cube, it’s going to make it unlikely that many people will find it as usable as I do. I probably should have got someone to play around with it for a while and give me feedback for stuff, because there are doubtlessly tons of little things that I’ve overlooked to make it more usable. For someone who is good at Rubik’s cubes and has a general knowledge of computer though, I think that they could use this program easily once they learn the controls.

For that I kind of wish I had pushed to get a better graphics library like openGL to run my pictocube, because it would have been unique and something different that could make solving it on the computer, well, different from solving it by hand. Or even getting the wiimote to control, may have given people a real challenge and something fun to do, but I’m happy with what I’ve done.