Conclusion and future plan(if more time then…)

Computational models are important tools in modern days, researchers and decision makers can utilize these models to aid them during research or make important decision. But these models may often receive poor testing, this is because its developers are often not trained in software testing, and the testing process can be very complex and time consuming. \\\*\\\*

Advance testing method are developed to assist testing, but these methods are often too complex for people who are trained in software testing. One of these tools is Causcumber, a tool that use the behaviour of model to determine if the model meet the developer’s expectation. But this tool still suffers from the same problem as other advance tool, where it maybe too complex for untrain user, therefore discourage people from using it. \\\*\\\*

A tool to improve user experience of Causcumber is created, this tool can automatically turn user’s input into the format that can be accepted by Causcumber, and auto generate parts of background file to reduce user’s workload. With this tool, user won’t need to learn the syntax required by Causcumber, and the line of code required user manually input is also reduced.\\\*\\\*

Causcumber is still a in development tool, and there might be changes in the future, and there’s also more to improve for the assist tool. First is in the viewing function of graphviz graph, in the current version, it’s size is limited by the some technical issues, and will affect user experience, parts of these is due some limitation of Kivy api, this will require some time to improve. \\\*\\\*

Second is editing scenario outline and scenario in feature file, more hints or the ability to select how the parameter going to change (For example, increase, decrease) from a menu can be implemented. This is to be decide since this function might limit the options in testing. \\\*\\\*

Third is to further reduce the need for user to program, namely the three background files. In those files, a similar like edit scenario in feature can be added, but since these files are a lot more complicated, the editing function for these files may not be as simple to implemented, the way user implements their model may also affect this process, which adds more complexity to the implementation of this function.\\\*\\\*

Overall, this project set out to make a complex tool become more approachable, despite some technical challenges, this project provides some interesting results for more future work on this subject. With more functions implemented, testing will be more approachable in the future.