Team: Samantha Rack and Lucinda Krahl
Project: Audacity
Target Platform: Virtual Machine running Ubuntu

GM9

We had considered modifying the main data structure (blockfiles) that is used in Audacity to store audio data in a tree in order to address delay with edit and draw tools on long files. We were looking to identify what was causing the access of the data in the blockfiles to slow down as the file size grew. However, after looking into the code and thinking about the centrality of this data structure to the code, we have decided that the blockfiles are either (1) optimized to their full potential (this would be something developers would have focused on since it is so important to all uses of Audacity), or (2) too involved for us to understand where the structure could be improved and what other files are depending on it.

Next, we looked into the UndoManager class, which is used to enable users to undo actions they have just completed. One of the member functions determines how large each Undo structure is. It populates a map structure with pointers to blockfile objects, and only adds to the map if the block has not already been referred to. The map is populated with the [ ] operator, and takes O(log n) time for the insertion (std::map is implemented as a red-black tree). However, if we were to replace this with a hash, then insertion would be average case O (1) time. Throughout the class, there are other uses of the map that could result in a performance improvement if a hash were used instead.

GM 10

For this milestone, I added a private variable and initialized it in the UndoManager class. The code still compiled and ran as expected. I checked that Undo still functioned correctly. There were no error messages generated, but the member variable did not affect any of the preexisting code.