1. **Briefly describe one important lesson about modularity that the paper describes and that you think is still relevant today.**

   I believe the hierarchical structure of the modules are still relevant today, especially in regards to having ‘dependencies’ and ‘uses’ based on a previously created module. This way, the other future (or later) modules can utilize one or many modules from the start (or level one). The perfect example given in the reading was the symbol table being used in other applications or in another project, and for us to not build a brand new module.

2. **Computing has advanced significantly since this paper was written. Briefly describe one challenge (not necessarily with modularity) or perspective that, while realistic in 1972, does not apply today.**

   It seems the system design or the application in the reading is mainly referring to a single solid state program or a project where the code and its mapping stay in one machine, possibly somewhere local. Nowadays, we have computing power across geographical regions where API calls, cloud based storage, network calls, and general internet based practices are utilized.