Rust Universal Machine ReadMe

Program Analyzers: Norman Winn and Nick Goltsos

Time to solve: 20 hrs

World of ideas contributors: Colin (TA), Arlen D, Connor G, Olivia E, Lynsie L, Dan M

We have implemented all portions of the universal machine. As far as we know, everything should work as intended but limited time constraints may have limited our ability to fully test the program. We haven't departed too far from our initial design but we have implemented concepts like information secrecy by outsourcing the instructions to an instructions.r file and having bitpacking as a separate file provided by Professor Daniels has that built in information secrecy. Speaking of which, we used Professor Daniels bitpacking as our last assignment went rather poorly and we wanted to ensure that bitpacking would work as intended off the bat so that we could get down to business. We have organized our architecture into five main files, a lib.rs file, the main.rs file, the instructions.rs (which contains the commands of all of the instructions), the vMachine.rs file (which contains the structure and implementation of the virtual machine itself), and bitpacking module provided by Professor Daniels as we described earlier. These modules help encapsulate the information needed by each file by limiting the amount of data within each file to strictly what is necessary. This helps keep some portions of the implementations secret which is a good security practice. It took us a good while to actually understand how to manage this program in representation. To put a rough number on it, we analyzed the assignment for roughly 2 hours, worked around 6 hours developing a plan of attack to create the virtual machine, and the rest of the time, roughly 12 hours, implementing the virtual

machine itself. We did run into a fair bit of issues and Connor Gray helped get us through it. He has been a critical force of guidance in our design and implementation processes and I appreciate his support.