Our experience with programming language exploration in Rust has been mostly positive. In general, the standard library is robust, and the language has encouraged us to program in ways that eliminate bugs before they appear. We determined early on that Jake had more experience with the language, making him a great candidate for the co-pilot role of pair programming. Being the pilot, Ryan was able to gain experience with common Rust conventions and language-specific design principles. As we moved forward throughout assignments, both partners spent time in both pair programming roles. We pair programmed every line of code for each assignment, and it both kept us productive and allowed us to learn quickly during our sessions.

For each assignment, we began by researching key parts of the assignment and determining if there was already a library that met the requirements. We quickly found the serde_json library for the xjson, and functions in the standard library for xyes and xtcp. However, we ran into issues with Rust's Cargo package manager when trying to find versions of packages for xgui which are compatible with the Khoury linux machines' older version of Rust. We tried over 20 different GUI libraries before we successfully compiled our project with an older version of gtk.

After adding the libraries with the functions we needed, it was simple to write the code needed to glue these functions together. Often we'd find that the libraries we used were explicit in the errors they returned and we could then choose to propagate them as well or handle them in place with a match expression. Overall, pair programming combined with prior research allowed us to effectively complete the exploratory assignments, and we thoroughly enjoyed working with the Rust language as we did so.