

Conclusion

TABLE OF CONTENTS

- 1 [Adding features from the Rust version](#)
 - 2 [MapReduce applications](#)
-

Congratulations on finishing the assignment! Here are some optional, not-for-credit things you can do to improve your MapReduce system.

These are only for fun; there is no extra credit provided.

Adding features from the Rust version

The [Rust version](#) of this assignment has heartbeats to avoid reassigning long-running tasks and in-memory buffers for storing intermediate results to make the cluster more realistic (intermediate results are usually not written to a global file system). Take a look at the Rust spec to see if you can incorporate these ideas into your implementation.

You can add additional RPCs to handle this, or modify your existing RPCs. You may want to take a look at the Rust RPCs as a starting point.

MapReduce applications

Try implementing your own MapReduce application! You can get ideas for what to implement from the MapReduce paper.

Look at `src/app/grep.rs` and `src/app/wc.rs` to see how to fit a MapReduce application into the framework we provide. You'll also need to modify `src/app/mod.rs` to allow workers to find your new application.
