Names: Phuc Nguyen (32593321) and Hai Nguyen (32610558)

Gigapaxos:

For the constructor, we initiated a Cassandra instance like in PA2. We also decided to store the keyspace name from the arguments so we may use it later when we want to retrieve the state of the application from Cassandra.

For execute(), we extracted the request value from the parameter and executed it in Cassandra. If we get a bad request then the program returns false, and GigaPaxos would attempt to rerun it anyway.

For checkpoint(), we first queried Cassandra to get every row of the table, then extracted the ID and events from the rows. We then appended them into a string called fullState, where it'd store every single operation to restore each row to a certain checkpoint. We'd then store this string to a hash map with the name of the checkpoint/application name/parameter being the key.

For restore(), we check that the supplied state is not empty first before executing every single command within that state.

We passed all of our tests. But when we did fail them, the primary reason was that we misunderstood how the checkpoint() and restore() were being called from GigaPaxos, thus we needed to modify how we were storing the state names and that we also had to re-execute queries up to a certain checkpoint in restore().