**Winston Zhang, wyz5rge**
**CS 4414**
**6. April 2023**
**p2exp2b**

In assignment p2exp2b, students were asked to create a scalable multithreaded hash table implementation. This short write-up details my attempts at achieving this goal.

My implementation is based on the boilerplate code provided in the git repository. This was done in the interest of ease of compilation.

The following changes were made:

- *find_which_hashtable* was modified to take in a key as a parameter, and return an integer that would be the function's selection of hashtable for the key-value pair to be found in
- *thread_func* was modified so that hashtable operations and the associated locking and unlocking would be performed with multiple hashtables allocated
- *main* was modified so that multiple hashtables are allocated, equal to the thread count

These changes greatly improve the scalability of the program, as thread count increased. This is the output of run.sh before changes:

p2-concurrency > exp2b-assignment > test_output > source.txt

```
./hashtable-biglock output/hashtable-mono.txt
test=hashtable threadNum=1 iterations=1000000 numList=1 numOperation=1000000 runTime(ms)=161 tput(Mops)=6.20
test=hashtable threadNum=2 iterations=1000000 numList=2 numOperation=2000000 runTime(ms)=1270 tput(Mops)=1.57
test=hashtable threadNum=4 iterations=1000000 numList=4 numOperation=4000000 runTime(ms)=3766 tput(Mops)=1.06
test=hashtable threadNum=6 iterations=1000000 numList=6 numOperation=6000000 runTime(ms)=6177 tput(Mops)=0.97
test=hashtable threadNum=8 iterations=1000000 numList=8 numOperation=8000000 runTime(ms)=10190 tput(Mops)=0.79
test=hashtable threadNum=10 iterations=1000000 numList=10 numOperation=10000000 runTime(ms)=17243 tput(Mops)=0.58
test=hashtable threadNum=12 iterations=1000000 numList=12 numOperation=12000000 runTime(ms)=28762 tput(Mops)=0.42
test=hashtable threadNum=16 iterations=1000000 numList=16 numOperation=16000000 runTime(ms)=21658 tput(Mops)=0.74
test=hashtable threadNum=20 iterations=1000000 numList=20 numOperation=20000000 runTime(ms)=29096 tput(Mops)=0.69
All tests done

```

and this is the same script's output after changes were made and the code was re