Report

I Introduction:

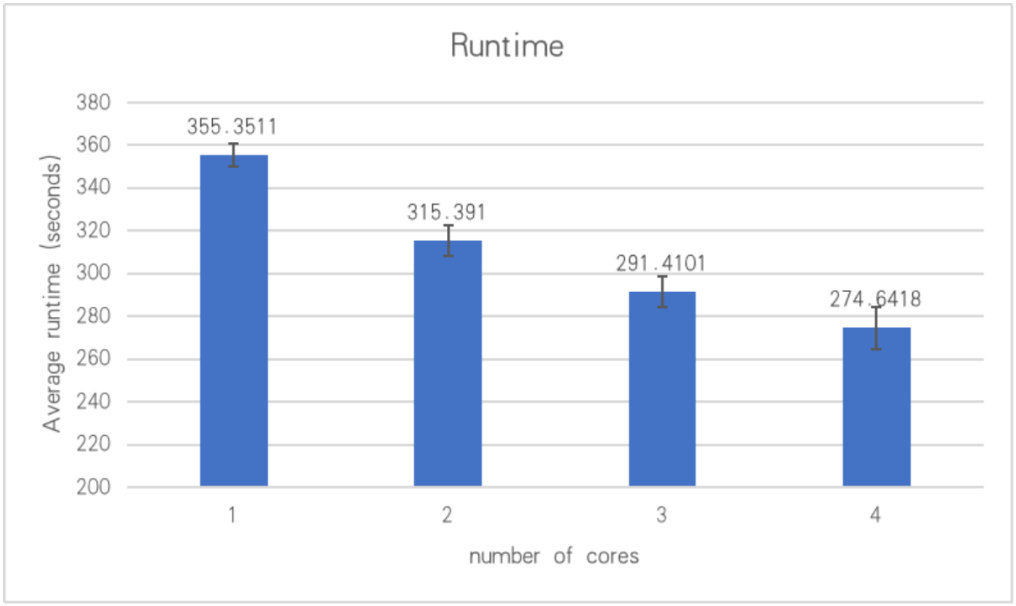
In this homework, we are testing the Scalability of a Stock-exchange program. This program is implemented in C++ and Postgresql. This program allows users to create accounts, add symbols and make transactions, check orders and cancel their opened transactions. The system will match the opened transactions and find the optimal deal price for the user. In conclusion, our program can help user exchange their shares and match the buyers' and sellers' requests.

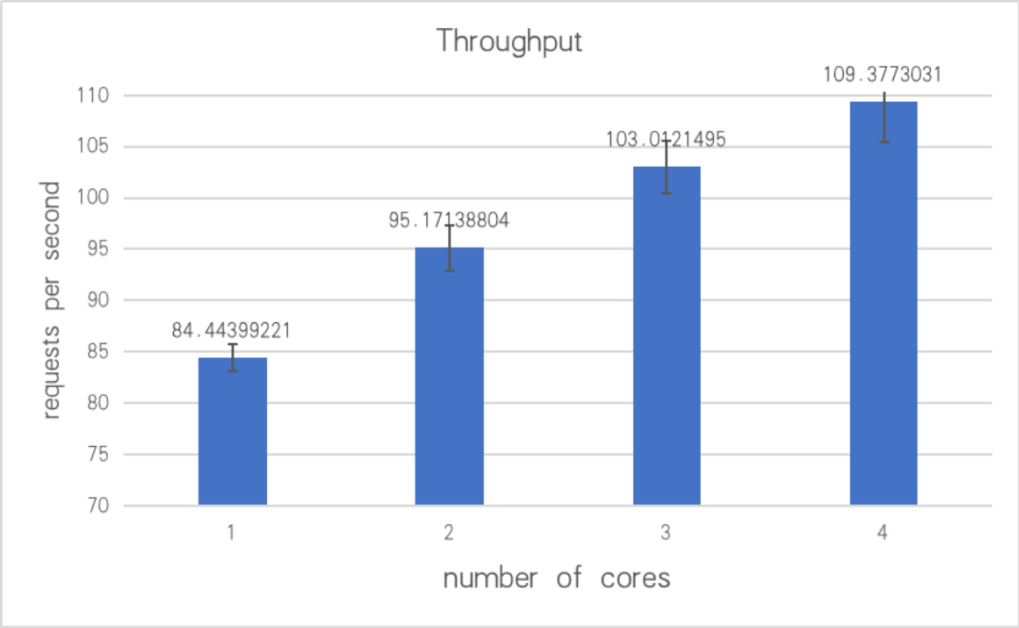
II Testing strategy:

For testing the scalability of our program, we use 2 strategies. In the first one, we set up only one client to send requests and the server will deal with the only client's request. In the second one, we test multiple clients running in parallel, and the server will deal with multiple clients' requests and the Data will be processed by the program and the database.

A. Single Client Test

In the Single Client test, we flood our server with 30000 requests through one client in serial. The program running time is recorded along with the throughput of the program. Below are the graphs:

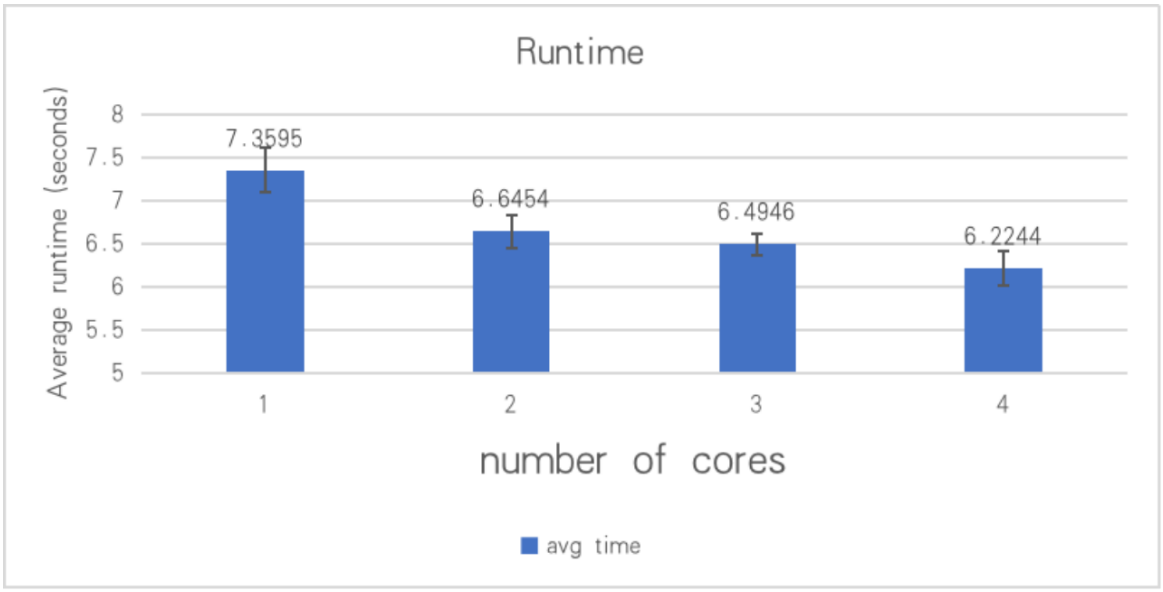


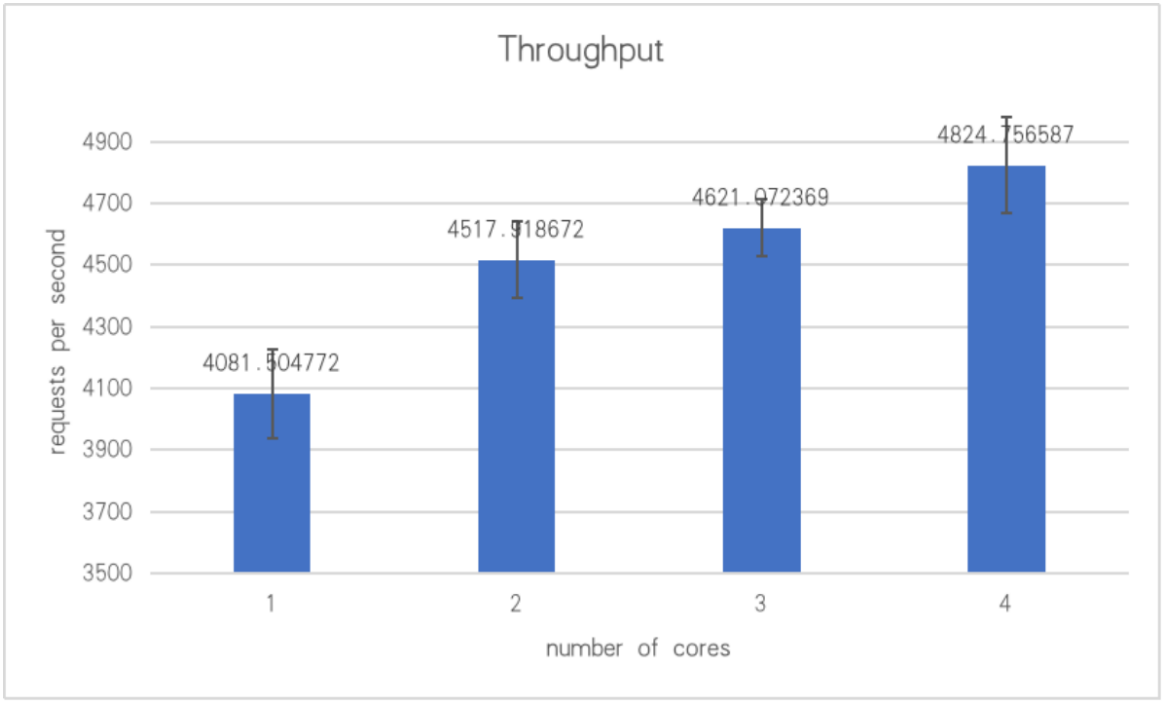


From the data figures, we can conclude that our program has good scalability: the more the number of cores is, the faster the program runs and the more the throughput is.

B. Multiple Clients Test:

In the Multiple Clients test, we send 30000 requests through 200 threads and each thread sends 150 requests in parallel. The program running time is recorded along with the throughput of the program. Below are the graphs:





From the data figures, we can conclude that our program enjoys good scalability under the condition of multiple cores with multiple threads. The performance of our program increases with the number of cores executing the program and it is indicated by the throughput and execution time. In general, the average runtime decreases as more cores are engaged and the throughput increases ideally, demonstrating a pretty good scalability of our program.

III : Problem log

Record noteworthy issues:

1. Above tests were run using ./server directly. After using docker-compose up command to run the tests, the performance gets worse and even comes up with connection failure or some other bugs sometimes, and the maximum number of threads and requests that can be processed properly has been reduced. We tried hard but are not able to fix this problem.
2. Please run “chmod o+x run.sh” before docker-compose up.