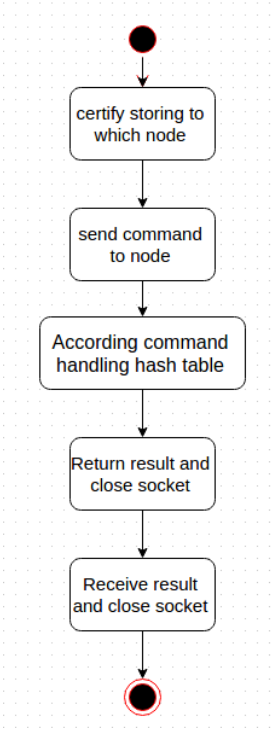
Design Document

System Design

A node process can be split to server part and client part.

When node set up, process starts server part to listen particular port, and client part to get command from users input. According command, client part sends command to particular port. When server part receive message from one node client part, server handle hash table in term of receive message. After operation for hash table, server parts of this node sends result to client parts of node, and close socket. Node's client part receives results, close socket and print result.

Hash table places in node's server part. The type of hash table is dictionary. Put operation is to update the dictionary. Get operation is to get value of the key. Del operation is to delete particular pair.



Synopsis Design

0 [KEY] [VALUE]	Insert value
1 [KEY]	Get value
2 [KEY]	Delete value
test_0 [KEY]	Evaluation_put
test_1 [KEY]	Evaluation_get
test_2 [KEY]	Evaluation_del

Node –help	Help document
e.g	
0 abc 123	Insert pair that key = 'abc' and value = '123'
1 abc	Get value which key = 'abc'
2 abc	Delete pair which key = 'abc'
test_0 abc	Insert 100000 pairs that key = 'abc0' ~ 'abc99999' and value = key
test_1 abc	Get 100000 values that key = 'abc0' ~ 'abc99999'
test_2 abc	Delete 100000 pairs that key = 'abc0' ~ 'abc99999'

Possible Improvement

Using asymmetric communication for each request.

Resilience of system.

Using more than one table in each node.