Homework 5

CS425/ECE428—Spring 2020

Due: 11:59 p.m. Wednesday, April 22

1. DHT . . . . . 13 points

Consider a Chord DHT with a 16-bit address space and the following 100 nodes. (Hexadecimal values in parentheses for easier manual finger computation).

```
  451  (1c3),   1195  (4ab),   1460  (5b4),   3854  (f0e),
 4175 (104f),   4400 (1130),   5651 (1613),   6357 (18d5),
 7483 (1d3b),   9105 (2391),  10073 (2759),  10483 (28f3),
10825 (2a49),  11273 (2c09),  11446 (2cb6),  11535 (2d0f),
11693 (2dad),  11925 (2e95),  12324 (3024),  13700 (3584),
15390 (3c1e),  15925 (3e35),  16030 (3e9e),  16569 (40b9),
16737 (4161),  17154 (4302),  19264 (4b40),  19477 (4c15),
19503 (4c2f),  19919 (4dcf),  19954 (4df2),  21065 (5249),
21324 (534c),  22507 (57eb),  22508 (57ec),  22972 (59bc),
23062 (5a16),  23168 (5a80),  23403 (5b6b),  23478 (5bb6),
23933 (5d7d),  24070 (5e06),  25209 (6279),  27513 (6b79),
29012 (7154),  30150 (75c6),  30179 (75e3),  30297 (7659),
31325 (7a5d),  32048 (7d30),  33824 (8420),  34612 (8734),
35080 (8908),  35394 (8a42),  35643 (8b3b),  37013 (9095),
37957 (9445),  38222 (954e),  39475 (9a33),  39558 (9a86),
39683 (9b03),  40571 (9e7b),  40996 (a024),  41553 (a251),
41995 (a40b),  43305 (a929),  43541 (aa15),  44307 (ad13),
44623 (ae4f),  45889 (b341),  46198 (b476),  46655 (b63f),
48044 (bbac),  48736 (be60),  49061 (bfa5),  49720 (c238),
50683 (c5fb),  52347 (cc7b),  52806 (ce46),  52841 (ce69),
53305 (d039),  53819 (d23b),  55639 (d957),  56470 (dc96),
56905 (de49),  57082 (defa),  57250 (dfa2),  57496 (e098),
57585 (e0f1),  58280 (e3a8),  58800 (e5b0),  59048 (e6a8),
60672 (ed00),  61532 (f05c),  61851 (f19b),  62819 (f563),
63086 (f66e),  63966 (f9de),  64459 (fbcb),  64654 (fc8e),
```

You can download these in JSON format from `https://dist.systems/assets/hw/hw5-nodes.json`

(a) (3 points) List the fingers of the node 22972 (59bc).

(b) (3 points) List the nodes that would be encountered on the lookup of key 17151 (42ff) by node 22972 (59bc)

(c) (2 points) Identify the nodes that will store the largest expected number of keys and the smallest. (Assume for now that a key is stored at only the successor node.) What is the ratio of their expected storage?

(d) (3 points) A power outage takes out half the nodes: the ones with odd identifiers. Assume no stabilization algorithm has had a chance to run, and so the finger tables have not been updated. List the nodes that 22972 (59bc) would contact to look up the key 17151 (42ff). (When a node in the normal lookup protocol tries to contact a finger entry that is no longer alive, it switches to the next best finger that is alive.)

(e) (1 point) How many successors does each node need to maintain to ensure that the ring does not become disconnected after the above power outage?

(f) (1 point) How many successors and predecessors need to host a replica of a key to ensure it does not get lost in this power outage?

2. RPC ..... 7 points

(a) (4 points) Using an Interface Description Language such as Apache Thrift, Avro, or Protocol Buffers, write an RPC specification for Paxos interaction between Proposers and Acceptors. (Ignore Learners.) You should define an RPC for a proposal and another for an accept request.

(b) (3 points) The Python list objects have the following methods defined on them `append`, `extend`, `remove`, `pop`, `clear`, `index`, `count`, `sort`, `reverse`, and `copy`, They are described here: `https://docs.python.org/3/tutorial/datastructures.html` Which of the methods are idempotent? Briefly explain your answer.

3. Concurrency ..... 10 points

Consider the following three transactions:

	$T1$	$T2$	$T3$
1	read X	read Y	read Z
2	read Y	read X	read Y
3	write Z	write Y	write X
4	write X	write Z	write Y

(a) (4 points) Write down all the conflicts between the operations in the three transactions. (You can refer $Tn.m$ to each operation; e.g., $T2.3$ is "write Y").

(b) (2 points) Write down a *non-serial* interleaving that could result from using *shared* two-phase locking and is equivalent to $T1$ followed by $T2$ followed by $T3$.

(c) (2 points) Write down a partial interleaving that would lead to deadlock when using *exclusive* two-phase locking.

(d) (2 points) Now consider these two transactions:

$T4$: read X; read Y; write X; write Y; write Z

$T5$: read X; write X; read Y; read Z; write Z

Write down an interleaving of $T4$ and $T5$ that is serially equivalent but impossible with either *shared* or *exclusive* two-phase locking.