1	2	3	4	5	6	7	8	Total
/8	/8	/8	/8	/8	/8	/8	/8	/60

LUMS School of Science and Engineering
Department of Computer Science

Final Exam – Spring 2013

CS 570 — Advanced Operating Systems

Roll no: _____

Instructions.

- YOU CAN CROSS OUT 4 POINTS WORTH OF QUESTIONS. IF YOU DO NOT, LAST 4 POINTS WORTH OF QUESTIONS WOULD NOT BE CHECKED.
- You have 3 hours to complete the exam. The maximum possible score is 60.
- The exam consists of 8 questions and 9 printed pages.
- It is an open book, open notes exam.
- If there is any confusion, write down your assumptions and proceed to answer the question.
- In questions where an implementation is required, use the language of your choice. Do not worry about access modifiers (public, private). Assume *simple* helper functions. Assume member method names of collection classes (List, Stack, etc.).
- Manage your time. You should not be spending more than 3 minutes for one mark to finish on time.

1. The Google file system

4 pts

```
class ChunkServer {
  class Chunk {
    int _id;
    File _datafile;
  } chunkList[];
  int maxserialnumber;
  List<Pair<int,string>> LRUbuffer;
  int distance(ChunkServer c); // returns network distance
  void data(int id, string data, List<ChunkServer> forwardTo);
  bool primaryWrite(int id, List<ChunkServer> secondaries);
  bool secondaryWrite(int id, int serialnumber);
}
```

Let the above be partial definitions of a GFS ChunkServer. Note that id is passed from the client to identify the data while serialnumber should be assigned by ChunkServer to order writes.

(a) (4 pts) Implement ChunkServer.data which buffers data and forwards data to secondaries optimizing network performance.

(b) (4 pts) Implement ChunkServer.primaryWrite which is passed the id of previously sent data and returns true if and only if it is successfully written on all replicas.

2 pts

2. Making geo-replicated systems fast as possible, consistent when necessary

(a) (2 pts) Consider a course registration system where each course has a fixed capacity and a list of registered and waitlisted students. If add and drop are blue operations, describe a situation where the class capacity requirement can be violated.

(b) (2 pts) In the above example, divide adding a course into two shadow operations and tell their appropriate colors.

(c) (2 pts) An eventually consistent system is periodically sampled for a very long period of time and we never find the system to be consistent. Why is that?

(d) (2 pts) What does it mean to say that an eventually consistent system does not have stable histories?

4 pts

3. MapReduce: simplified data processing on large clusters

When you visit a profile on Facebook, you see a list of mutual friends. It would be wasteful to calculate this list in real-time. We would like to write a map-reduce job (in language of your choice) that takes input data in the form (friend,list_of_friends) i.e. the key is friend-id while value is list of friend-ids and outputs (pair_of_friend1_friend2, list_of_mutual_friends).

(a) (4 pts) Implement map

(b) (4 pts) Implement reduce

4. Bigtable: A Distributed Storage System for Structured Data

4 pts

```
class SSTable {
  int redo-point;
  void add(string key, string value);
  string readLast(string key);
}
class GFSFile {
  int lastRedoPoint;
  int append(string key, string value); // returns record number aka redoPoint
  Pair<string,string> read(int redoPoint);
}
class TabletServer {
  GFSFile _tabletLog; // most up-to-date log
  SSTable _sstables[]; // _sstables[0] is newest
  Dictionary<string,string> _memTable; // quick lookup for reading
  int _memTableThreshold; // _memTable should not grow bigger than this

  void write(string key, string value);
  void recover(GFSFile tabletLog, SSTable sstables); // tables ordered by time
}
```

Implement the following functions. You can use any language (C++, Java, Python). Ignore timestamps and assume tablet servers store key value pairs i.e. you do not worry about the mapping of row, columnfamily, column, timestamp, and value to a key value pair. You can assume *simple* helper functions.

(a) (4 pts) TabletServer.write

(b) (4 pts) TabletServer.recover

5. Practical, transparent operating system support for superpages

0 pts

(a) (2 pts) Under what circumstances a write to a superpage would cause a demotion?

(b) (2 pts) Under what circumstances a write to a superpage would cause a promotion?

(c) (4 pts) Given the following definition of a PopulationMap, implement a maxReservable function that finds the largest page size reservable containing the startingAddress passed to this function. It should return the page size reserable.

```
class PopulationMap {
  class Node {
    int somepop, fullpop;
    int pagesize;
    bool isLeaf;
    Node children[];
  }
  Node root;
  int startingAddress; // start addr of range rep. by this population map
  int maxReservable(int startingAddress); // returns largest reservation possible
}
```

0 pts

6. Virtual memory primitives for user programs

(a) (2 pts) In a shared virtual memory system, a page fault comes because a process wants to write to a page, and is forwarded to the user-space page fault handler. List the steps taken by the user-space handler.

(b) (2 pts) How does concurrent garbage collection allow the collector to perform garbage collection on a page while prohibiting concurrent access from mutators.

(c) (2 pts) In incremental checkpointing, the process is interrupted for some setup work and then resumes and works concurrently with checkpointing. What happens if the process tries to modify data that is not yet saved in the checkpoint?

(d) (2 pts) Why shouldn't the TLB be flushed when a read-only page is made read-write? What happens if a write request comes? Would that be denied?

7. Exokernel: an operating system architecture for application-level resource management

2 pts

(a) (2 pts) List two key differences between an exokernel and a microkernel.

(b) (2 pts) How is visible revocation different from revocation in normal operating systems. Explain with an example of processor time slice.

(c) (2 pts) What does it mean to say that time slices can be allocated in a manner similar to physical memory?

(d) (2 pts) Since Aegis does not implement demand paging, how can we load a program in memory thats larger than available memory?

8. Xen and the art of virtualization

2 pts

(a) (2 pts) How are shadow page tables used in full virtualization?

(b) (2 pts) How are machine page tables protected from guests in Xen's para-virtualization?

(c) (2 pts) List one similarity and one difference between Xen and an exokernel.

(d) (2 pts) Whats the difference between a hypercall and syscall?

Good Luck!