EXAM 2019

1.-

a) F. In case of reads, each replica replies to the client. The proxy being used in clients should be aware of read requests and forward them to a single replica. (Optional optimizations)

b) F. Consistency is, at least, sequential, and it doesn’t improve service throughput.

**c) T. All replicas behave in the same way. A single failure is easy tolerated.**

d) F. The replication model doesn’t vary the deployment needs.

2.-

**a) T. A new primary should be chosen and in the interim new requests may be lost.**

b) F. It is a requirement in the active model, but not in the passive one.

c) F. That was a requirement in the multi-master replication model in order to achieve eventual consistency.

d) F. It tolerates it.

3.-

a) F. There may be scenarios of a complete failure (disasters).

b) F. All failures are the result of a previous error, that should have been preceded by a fault.

c) F. Not necessarily “always”. For instance, with redundancy many errors don’t become failures.

**d) T.**

4.-

**a) T.**

b) F.

c) F.

d) F.

5.-

a) F. It needs an ACK to update.

**b) T.**

c) F. It can provide eventual consistency, but not sequential consistency.

d) F.

6.-

a) T

**b) F. It only gets eventual consistency. And eventual consistency is more relaxed than sequential consistency.**

c) T.

d) T.

7.-

a) F. It doesn’t specify anything about consistency.

b) F. Docker doesn’t automate adaptability decisions.

**c) T. This matches the definition of elasticity.**

d) F. This doesn’t fulfil the condition of scalability nor adaptability.

8.-

a) F. We are renouncing to availability.

b) F. We are renouncing to availability.

c) F. Every process goes on.

**d) T.**

9.-

**a) T. All workers processes share the same part of the program. In that node an instance of a bind() operation is done. So, all they share same port.**

b) F. To write adaptive program is very easy, but it is the responsibility of the programmer.

c) F. They don’t share memory. They use separate processes.

d) F. All those processes run in the same computer.

10.-

a) F. MongoDB needs passive replication

**b) T.**

c) F. It doesn’t use transactions.

d) F. It needs passive replication.

11.-

a) F.

b) F.

**c) T.**

d) F.

12.-

a) F.

**b) T.**

c) F.

d) F.

13.-

a) F.

**b) T.**

c) F.

d) F.

14.-

a) F.

b) F.

**c) T.**

d) F.

15.-

a) F.

**b) T.**

c) F.

d) F.

16.-

a) F.

**b) T.**

c) F.

d) F.

17.-

a) F.

b) F.

c) F.

**d) T.**

18.-

**a) T.**

b) F.

c) F.

d) F.

19.-

a)

b)

c)

d)

20.-

a)

b)

c)

d)