Assignments for week 13

Anders Degn Lapiki, Jacob Kjærulff Furberg, Jonas Ishøj Nielsen

19/11/2020

Contents

1 13.1

1.1 13.1.1 - green

Done

1.2 13.1.2 - green

1: commands:

```
git checkout -b newnumbers
change file: numbers.txt: 1234
git commit -am "first"
git push

git checkout master
git merge newnumbers
```

No conflicts.

```
2: commands:

git checkout newnumbers
change file: numbers.txt: 12312213
git commit -am "second"
git push

git checkout master
change file: numbers.txt: abe
git commit -am "second"
git push
git merge newnumbers
```

Merge conflict.

1.3 13.1.3 - green

Can't make the transformation for the write operations to keep both changes and instead it leaves it up to the user/us to decide.

1.4 13.1.4 - yellow

```
git checkout -b newnumbers
change file: numbers.txt: 12xy54p.
git commit -am "third"
git push
```

```
git checkout master
git rebase newnumbers --skip
```

Error is that it tries to put master's last commit in front of a sequence of commits containing a commit from master.

2 13.2

2.1 13.2.1 - green

1: one change is acceded first –> second change happens.
2: Alice and bob receive changes until they have gotten same number of operations as server have.
3: Alice push up.
4: Alice get acknowledgment –> Bob get changes.
5: Bob push up.
6: Bob get acknowledgment –> Alice get changes.
7: Alice push up.
8: Bob push up.
9: Alice get acknowledgment/transformation, Bob get acknowledgment/transformation –> "This is a short explanationThis explains".

2.2 13.2.2 - yellow

10: start from step 6
11: Bob enter "This ex"
12: Alice enter "This is"

If Alice push then bob push: becomes: "Introduction 'newline' This exThis ex"
If Bob push then Alice push: becomes: "Introduction 'newline' This exThis is"
if Alice push then bob pull and then push: "Introduction 'newline' This ex 'newline' This is"

3 13.3

3.1 13.3.1 - green

Web server, e.g. if bad connection

3.2 13.3.2 - green

Lan system, if connected though a switch which fails.

3.3   13.3.3 - green

Strict consistency = A write to a variable by any thread needs to be seen instantaneously by all other threads.
Strong eventual consistency = Don't get up-to-date values, but always get a valid snapshot.

Example: reading size of stripedWriteMap.

4   13.4

4.1   13.4.1 - green

See `WingStructue` in `Wing.java`.

4.2   13.4.2 - yellow

See `WingBuffer` in `Wing.java`.
Test succeeds

4.3   13.4.3 - Yellow

See `WingBufferAtomic` in `Wing.java`.
The `WingStructureAtomic` works by having a the list of elements be of size 10.000 as otherwise we would have to at each inc() create a new Atomic Array, which would require locking.

Test succeeds