

CSE 15L lab report 4

Log into ieng6

```
79250@Alice MINGW64 ~/Desktop/CSE15/cse15l-lab-report4
$ ssh cs151fa23gc@ieng6.ucsd.edu
```

- type `ssh` and copy the url from WEEK2's note.
- pressed `←` until the cursor move to `zz`.
- replace `zz` to `gc`.
- press `<enter>` to connect ieng6 server.

```
Last login: Fri Nov 17 21:53:09 2023 from 128.54.150.132
Hello cs151fa23gc, you are currently logged into ieng6-202.ucsd.edu

You are using 0% CPU on this system

Cluster Status
+-----+
| Hostname | Time      | #Users | Load | Averages |
+-----+
| ieng6-201 | 22:55:01 | 15     | 20.37 | 21.79, 22.93 |
| ieng6-202 | 22:55:01 | 15     | 5.08  | 4.44, 3.52  |
| ieng6-203 | 22:55:01 | 10     | 4.54  | 4.70, 7.09  |
+-----+

Fri Nov 17, 2023 10:57pm - Prepping cs151fa23
[cs151fa23gc@ieng6-202]:~:230$
```

Clone your fork of the repository from your Github account (using the SSH URL)

```
[cs151fa23gc@ieng6-202]:~:220$ git clone git@github.com:SouCoder3/cse-15l-lab7.git
Cloning into 'cse-15l-lab7'...
remote: Enumerating objects: 68, done.
remote: Counting objects: 100% (34/34), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 68 (delta 20), reused 18 (delta 16), pack-reused 34
Receiving objects: 100% (68/68), 378.54 KiB | 1.83 MiB/s, done.
Resolving deltas: 100% (26/26), done.
[cs151fa23gc@ieng6-202]:~:221$
```

- type `git clone` and copy and paste the url from my fork repository. press `<enter>`.

Run the tests, demonstrating that they fail

```
Receiving deltas: 100% (26/26), done.
[cs151fa23gc@ieng6-202]:~:221$ cd cse-15l-lab7/
[cs151fa23gc@ieng6-202]:cse-15l-lab7:222$ bash test.sh
JUnit version 4.13.2
..E
Time: 0.536
There was 1 failure:
1) testMerge2(ListExamplesTests)
org.junit.runners.model.TestTimedOutException: test timed out after 500 milliseconds
    at ListExamples.merge(ListExamples.java:44)
    at ListExamplesTests.testMerge2(ListExamplesTests.java:19)

FAILURES!!!
Tests run: 2, Failures: 1
[cs151fa23gc@ieng6-202]:cse-15l-lab7:223$
```

- type `cd` and then type `cse<tab>`. It autofill the folder name. press `<enter>` to change the current work directory to `cse-15l-lab/`
- type `bash` and then type `tes<tab>`. It autofill the bash file name. press `<enter>` to run the test.

Edit the code file to fix the failing test

```
[cs151fa23gc0ieng6-202]:cse-151-lab7:223$ vim ListExamples.java
import java.util.ArrayList;
import java.util.List;

interface StringChecker { boolean checkString(String s); }

class ListExamples {

    // Returns a new list that has all the elements of the input list for which
    // the StringChecker returns true, and not the elements that return false, in
    // the same order they appeared in the input list;
    static List<String> filter(List<String> list, StringChecker sc) {
        List<String> result = new ArrayList<>();
        for(String s: list) {
            if(sc.checkString(s)) {
                result.add(0, s);
            }
        }
        return result;
    }

    // Takes two sorted list of strings (so "a" appears before "b" and so on),
    // and return a new list that has all the strings in both list in sorted order
    static List<String> merge(List<String> list1, List<String> list2) {
        List<String> result = new ArrayList<>();
        int index1 = 0, index2 = 0;
        while(index1 < list1.size() && index2 < list2.size()) {
            if(list1.get(index1).compareTo(list2.get(index2)) < 0) {
                result.add(list1.get(index1));
                index1 += 1;
            }
            else {
                result.add(list2.get(index2));
                index2 += 1;
            }
        }
        while(index1 < list1.size()) {
            result.add(list1.get(index1));
            index1 += 1;
        }
        while(index2 < list2.size()) {
            result.add(list2.get(index2));
            // change index1 below to index2 to fix test
            index1 += 1;
        }
        return result;
    }
}

"ListExamples.java" 50L, 1435C
```

```
import java.util.ArrayList;
import java.util.List;

interface StringChecker { boolean checkString(String s); }

class ListExamples {

    // Returns a new list that has all the elements of the input list for which
    // the StringChecker returns true, and not the elements that return false, in
    // the same order they appeared in the input list;
    static List<String> filter(List<String> list, StringChecker sc) {
        List<String> result = new ArrayList<>();
        for(String s: list) {
            if(sc.checkString(s)) {
                result.add(0, s);
            }
        }
        return result;
    }

    // Takes two sorted list of strings (so "a" appears before "b" and so on),
    // and return a new list that has all the strings in both list in sorted order.
    static List<String> merge(List<String> list1, List<String> list2) {
        List<String> result = new ArrayList<>();
        int index1 = 0, index2 = 0;
        while(index1 < list1.size() && index2 < list2.size()) {
            if(list1.get(index1).compareTo(list2.get(index2)) < 0) {
                result.add(list1.get(index1));
                index1 += 1;
            }
            else {
                result.add(list2.get(index2));
                index2 += 1;
            }
        }
        while(index1 < list1.size()) {
            result.add(list1.get(index1));
            index1 += 1;
        }
        while(index2 < list2.size()) {
            result.add(list2.get(index2));
            // change index1 below to index2 to fix test
            index1 += 1;
        }
        return result;
    }
}

index1
```

- type `vim` and then type `List<tab>.java` to open `ListExamples.java` in vim.

```
// and return a new list that has all the strings in both list in sorted order
static List<String> merge(List<String> list1, List<String> list2) {
    List<String> result = new ArrayList<>();
    int index1 = 0, index2 = 0;
    while(index1 < list1.size() && index2 < list2.size()) {
        if(list1.get(index1).compareTo(list2.get(index2)) < 0) {
            result.add(list1.get(index1));
            index1 += 1;
        }
        else {
            result.add(list2.get(index2));
            index2 += 1;
        }
    }
    while(index1 < list1.size()) {
        result.add(list1.get(index1));
        index1 += 1;
    }
    while(index2 < list2.size()) {
        result.add(list2.get(index2));
        // change index1 below to index2 to fix test
        index1 += 1;
    }
    return result;
}
```

- ```
// and return a new list that has all the strings in both list in
static List<String> merge(List<String> list1, List<String> list2)
{
 List<String> result = new ArrayList<>();
 int index1 = 0, index2 = 0;
 while(index1 < list1.size() && index2 < list2.size()) {
 if(list1.get(index1).compareTo(list2.get(index2)) < 0) {
 result.add(list1.get(index1));
 index1 += 1;
 }
 else {
 result.add(list2.get(index2));
 index2 += 1;
 }
 }
 while(index1 < list1.size()) {
 result.add(list1.get(index1));
 index1 += 1;
 }
 while(index2 < list2.size()) {
 result.add(list2.get(index2));
 // change index1 below to index2 to fix test
 index1 += 1;
 }
 return result;
}

}
```
- INSERT --

- ```
while(index2 < list2.size()) {
    result.add(list2.get(index2));
    // change index1 below to index2 to fix test
    index2 += 1;
}
return result;
```

- ```
// and return a new list that has all the strings in both list in sorted
static List<String> merge(List<String> list1, List<String> list2) {
 List<String> result = new ArrayList<>();
 int index1 = 0, index2 = 0;
 while(index1 < list1.size() && index2 < list2.size()) {
 if(list1.get(index1).compareTo(list2.get(index2)) < 0) {
 result.add(list1.get(index1));
 index1 += 1;
 }
 else {
 result.add(list2.get(index2));
 index2 += 1;
 }
 }
 while(index1 < list1.size()) {
 result.add(list1.get(index1));
 index1 += 1;
 }
 while(index2 < list2.size()) {
 result.add(list2.get(index2));
 // change index1 below to index2 to fix test
 index2 += 1;
 }
 return result;
}
```

- type `:wq<enter>` to save and quit vim.

## Run the tests, demonstrating that they now succeed

```
[cs151fa23gc@ieng6-202]:cse-151-lab7:224$ bash test.sh
JUnit version 4.13.2
..
Time: 0.015

OK (2 tests)

[cs151fa23gc@ieng6-202]:cse-151-lab7:225$
```

- type `↑↑` to access the bash history and get the `bash test.sh` command.
- press `<enter>` to run the test and get success.

## Commit and push the resulting change to your Github account (you can pick any commit message!)

### git add

```
[cs151fa23gc@ieng6-202]:cse-151-lab7:225$ git add *
The following paths are ignored by one of your .gitignore files:
ListExamples.class
ListExamplesTests.class
StringChecker.class
hint: Use -f if you really want to add them.
hint: Turn this message off by running
hint: "git config advice.addIgnoredFile false"
```

- `git add *` to add all change in the working directory to the staging area.
- The files methioned in `.gitignore` file would be ignore when add.

### git commit

```
[cs151fa23gc@ieng6-202]:cse-151-lab7:226$ git commit -m "Debug index"
[main 246793b] Debug index
Committer: Jiawei Huang <cs151fa23gc@ieng6-202.ucsd.edu>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

 git config --global --edit

After doing this, you may fix the identity used for this commit with:

 git commit --amend --reset-author

1 file changed, 1 insertion(+), 1 deletion(-)
```

- `git commit -m "Debug index"` : submit changes in the staging area to the local repository, and add message "Debug index" .

### git push

```
[cs151fa23gc@ieng6-202]:cse-151-lab7:227$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (1/1), done.
Writing objects: 100% (3/3), 298 bytes | 298.00 Kib/s, done.
Total 3 (delta 2), reused 2 (delta 2), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:SoulCoder3/cse-151-lab7.git
 10c2259..246793b main -> main
[cs151fa23gc@ieng6-202]:cse-151-lab7:228$
```

- `git push` : submit changes in local repository to remote repository.

```
while(index2 < list2.size()) {
 result.add(list2.get(index2));
 // change index1 below to index2 to fix test
 index2 += 1;
}
```

- double check the change in remote repository.