CSE 332: Data Structures and Parallelism — Spring 2020

Common CSE 332 Debugging Issues

General

The following items are issues you might run into during any (or all!) of the projects in CSE 332.

The IDE Debugger doesn't work with the provided tests

This happens because the testing framework does some fancy things to make sure different tests don't interfere with each other. If you want to use the debugger, copy the actual content of the tests to your own testing file and use that `main` instead. `sys.out.println` will also not work but you can use `sys.err.println` to print to console.

I keep on getting a `ClassCastException`

If the error is something like

```
java.lang.ClassCastException: [Ljava.lang.Object; cannot be cast to [Ljava.lang.Comparable,
```

this means that when you are creating an array (probably the one backing your data structure), you are using the wrong type. Take a look at the generics handout, and make sure that you're following it exactly.

I'm updating the size/root/etc., but it's not changing

It's likely that you're "shadowing" a variable. The super classes you're given have fields for `size`, `root`, etc. If you create your own in the subclasses, then your data structure has two conflicting roots. Make sure to use the one in the super class instead.

The tests are failing, but I'm pretty sure my data structures are working!

Make sure you're updating the `size` for your data structure. Also, make sure you're throwing the right exceptions based on what the super class says to do.

The tests time out on my computer, but not on gitlab-ci

The machine that we will be testing your code on is similar to the gitlab runners. Your personal machine may be significantly less powerful, and that's okay.

The tests fail on gitlab-ci due to a `ClassNotFoundException`, but not on my computer

Scroll to the top of the static analysis output on gitlab; it should indicate what the issue is and how to fix it. (The IDE compiler is more lenient about generics than `javac`, which is what the gitlab runners use.)

`Zip` or `uMessage` performs poorly or you are getting a `java.lang.OutOfMemoryError`

When you use the `new` keyword in Java, you are allocating memory on the heap. (Take CSE 351 for more information about memory allocation.)

(1) On the **Help menu**, click **Edit Custom VM Options**.

(2) Set the -Xmx option to 3G without the quotation marks, so the final line should be **-Xmx3G**

(3) Restart IntelliJ

(4) Go to **Run** | **Edit Configurations** | **Application** | **uMessage**

(5) For the **VM options**, add **-Xmx3G**

(6) Apply and press **OK** The above error occurs when your program allocates more memory than given by IntelliJ.

P2

testRepeatedWordsPerNGram is failing, but 3 == 3

You're likely creating a "new Integer", but you should be using the one you were given.

Cannot instantiate the type Dictionary<AlphabeticString, Integer>

Dictionary is an abstract class. So, you can't make a new one (which type of dictionary would it be?). You should use newInner and newOuter as if they were constructors for dictionaries. If you carefully read the spec, it explains exactly how to do this.

NullPointerException in MoveToFrontList

Make sure you're using .equals instead of ==

NullPointerException in NGramToNextChoicesMap

Make sure that your `HashTrieMap` is updating its `size` field correctly when changing the value of items that already exist in the `HashTrieMap`.

The Iterator for BST (or AVLTree) seems broken

Your worklists should be capable of storing nulls.

The type E is not a valid substitute for the bounded parameter <E extends Comparable<E> > of the type MinFourHeap<E>

You forgot to remove the "extends Comparable<E>" from the top of your MinFourHeap.

My ChainingHashTable puts identical items in the dictionary multiple times

Remember that hash tables rely on the equals method; you should make sure you're comparing the keys, not the items. Also, make sure you're using .equals and not ==

HeapSort and TopKSort pass the tests locally but not on gitlab

Make sure your MinFourHeap is initializing arrays of Object rather than Comparable.

My ChainingHashTable passes the gitlab tests, but when I run uMessage on a large corpus, it doesn't work

Make sure that you're handling the case where the table gets really big–the corpus might be larger than the sizes you've provided.

uMessage gives a NullPointerException when I type really fast and it tries to autocorrect

Don't worry about this; it's not your fault.

P3

Minimax times out on gitlab-ci!
Make sure that you're not computing things you don't need to. For example, if you hit the base case: do you need to compute all the moves? Do you need to copy the board?

When I run my bots, I get a `ConcurrentModificationException`
The order that you copy a board, make a move on that board, and recurse is very important. You are likely modifying the same board in two threads. This can happen if you're not copying it in one of the threads (e.g., the one you "compute" on). It can also happen if you make a move before copying.

My bot is returning moves that don't appear in `board.generateMoves()`
Make sure that you always return a move from `board.generateMoves()` (which contains all valid moves for the current player), bearing in mind that `board.applyMove()` changes which player's turn it is.

My bot appears to be running forever (even after increasing the timeout)
Make sure your code handles the case where `generateMoves()` yields a list of size 0 or 1, is not trying to `join()` any `compute()`'d tasks, and is not passing arguments in the wrong order (e.g. passing `hi` as `depth`).

My bot passes the tests, but it can't beat `calculon`!
Make sure you've increased the search depth in `Engine.java`. Also make sure that your base cases (particularly `moves.isEmpty()`) are correct, otherwise your bot may deliberately avoid winning.

`TestGame` doesn't stop at a stalemate!
Yes. This is expected behavior. If you reach a stalemate, you can stop the program. `TestGame` is meant for testing–not for actual playing.