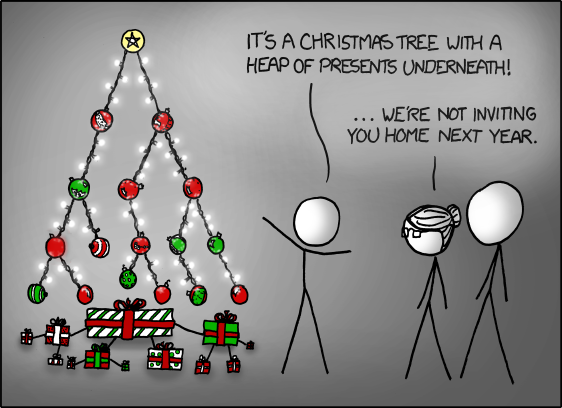
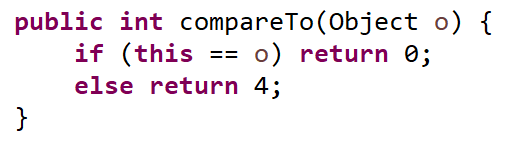
Instructions and Rules:

* You are responsible for answering all 22 questions in this exam
* You may write directly in the doc file or on the PDF file or on separate paper. Please try to include all your work in one file (you can photo and paste handwriting into this doc file, or you can code in Eclipse and snip+paste into this doc file, for example).
* You may use all resources available to you, including:
  + Previous labs
  + Previous exams
  + Previous quizzes
  + Notes, lecture notes/recordings
  + Previous written assignments
  + Eclipse or other Programming Development Environments
  + The textbook
  + The world wide web
* IMPORTANT: If you are using an answer (directly or adapting) from any source, cite the source that you are adapting your solution from. Failure to cite your work will result in harsh penalties of a grade of 0 on this exam and possibly in the course and possibly suspension from the college.
* Yes, you may use code and answers you find online to answer these questions. Again, your answers MUST be cited if they are not your own work.
* Yes, you may search StackOverflow, but you are NOT permitted to ask these questions from the exam to get an answer from another person.
* To reiterate: you are **NOT allowed** to ask another **person** or ask a **forum** about any part of this exam, and submitting this exam is understood to mean that any uncited content of this exam is entirely your own work.
* You will NOT have enough time to websearch to figure out how to solve a problem. If you encounter a problem you cannot make progress on, it is suggested you move on to a problem that you can effectively spend your time on.
* This exam runs from 1pm to 4pm. You must submit by 4pm.
* Good luck! And don’t forget: this exam is only worth 20%

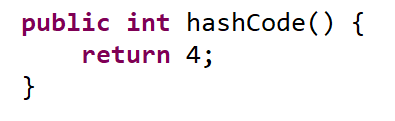


1. (2 points) Given the following code for a compareTo() function in a comparable object, determine if it is a valid compareTo() method as prescribed in the Java documentation for compareTo():



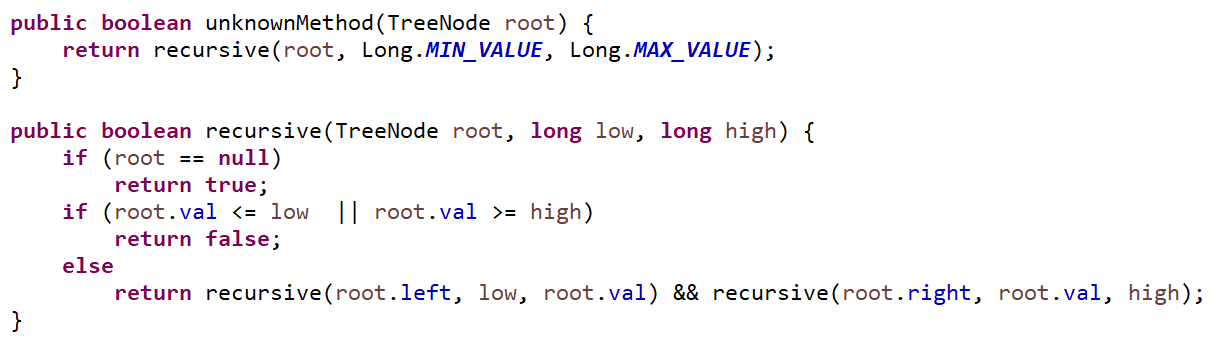
Justify your answer with one or two sentences.

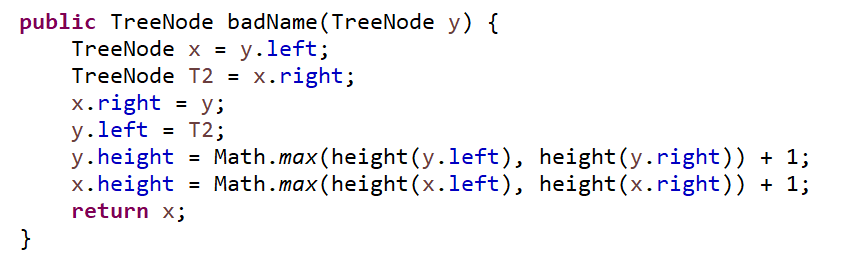
1. (2 points) Given the following code for an object’s hashCode() function, determine if it is a valid hashCode() method as prescribed in the Java documentation for hashCode():

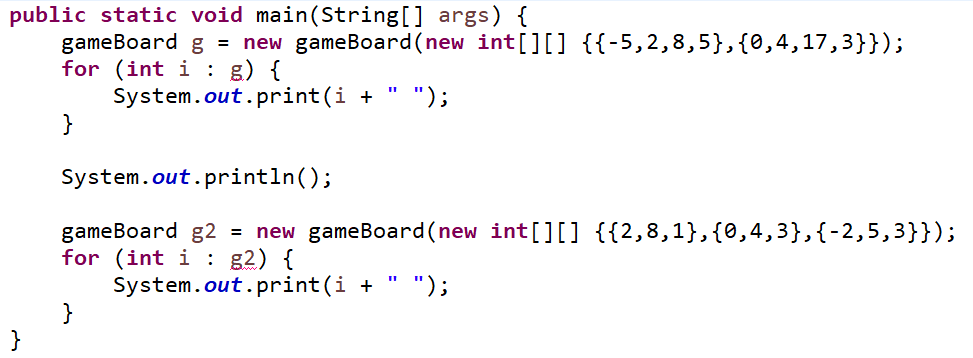


Justify your answer with one or two sentences.

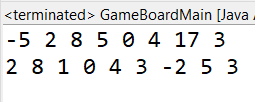
1. (3 points) The following two methods are named poorly. Either explain in 2-3 sentences what this code is doing or choose some descriptive names for these methods and give 1-2 sentences of justification for your choices.



1. (2 points) Give an appropriate name for the following method, with 1-2 sentences of justification. 
2. (5 points) Here is a main method that attempts to instantiate some game boards of various dimensions, and then attempts to iterate over the elements of their respective game boards with for-each loops.

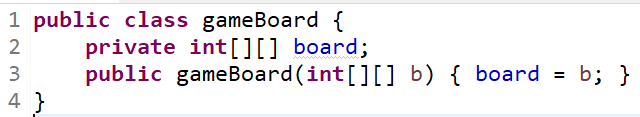


We would like the output of this main method to be as follows:

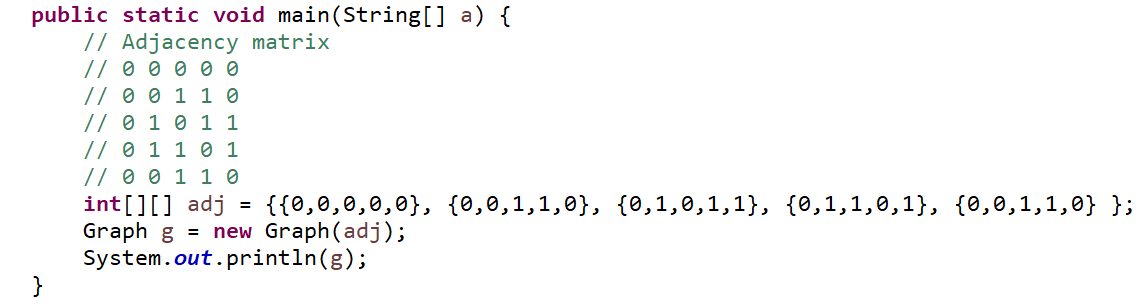


The underlined errors in the code above state: “Can only iterate over an array or an instance of java.lang.Iterable”. Make the above code work without modifying the above code, and only by making the gameBoard class Iterable.

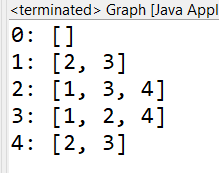
The current code for gameboard is as follows, and it is provided to you in the exam files on Moodle.



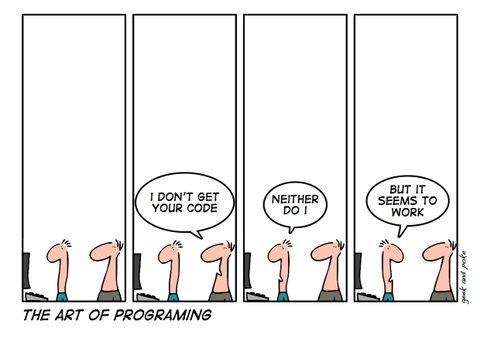
1. (5 points) The Graph.java file from lab8 is given in the final exam files. Add a constructor to the Graph.java file which accepts an adjacency matrix as parameter and instantiates the graph with the corresponding nodes and edges. Since the graph data structure is set up for vertex objects to be Strings, given an n x n adjacency matrix, you can simply name the n vertices of the graph with the strings “0”, “1”, … “(n-1)”. Here is an example instantiation with a 5x5 adjacency matrix:



The output from the above main method would be:



Your constructor method may assume that the given matrix will always be square (equal number of rows and number of columns) and will only contain 0s and 1s in a consistent manner (i.e. you do not have to implement any error-checking).

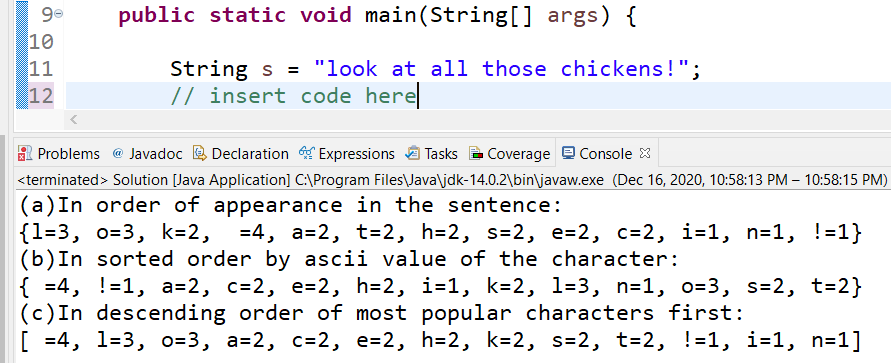


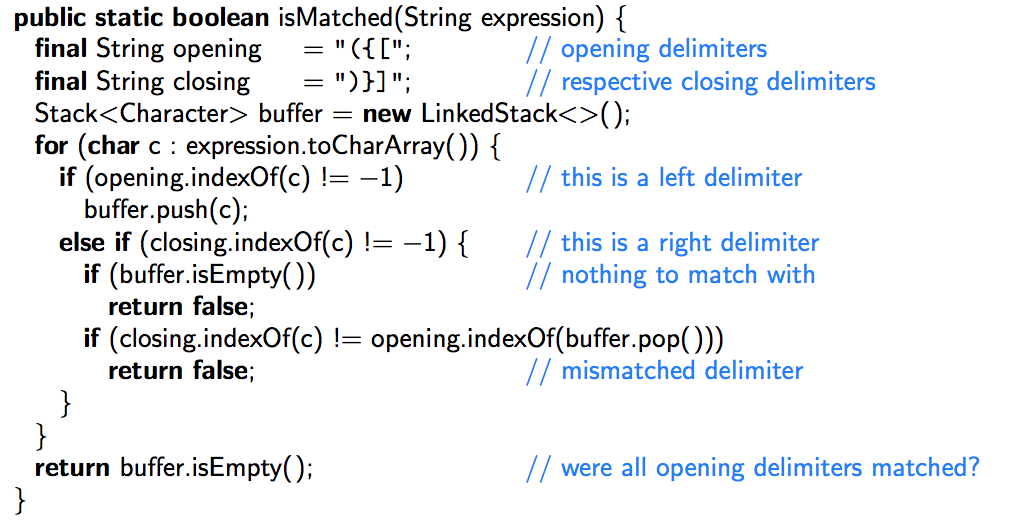
1. Given a String of characters, write some code that will produce a listing of the characters and how often the characters appear in the string, where the output is:  
   (a) (2 points) sorted in order of when the character appears in the sentence

(b) (2 points) sorted in order of the ASCII value of the characters

(c) (2 points) sorted in order of the most common characters first, down to the least common characters.

Here is an example of the expected output. You can ignore any kind of brackets or braces ( [ ] { } ) in your output.



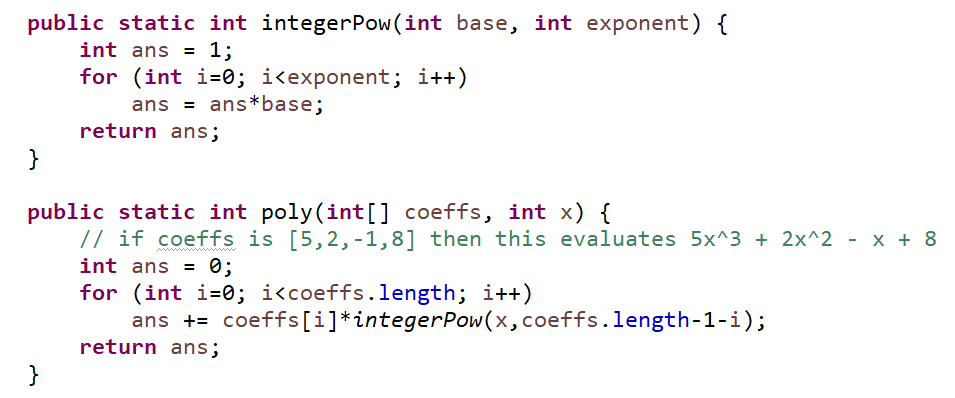
8. (4 marks) Consider the code on the right. With the expression

[2[[x+y{2-[z]}])

as input, write the contents of the stack when c=’z’ and also when the method terminates.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| When  c = z |  | At the end |

1. Consider the following code:



1. (2 points) If coeffs is an array of length n, what is the Big-O runtime (in terms of n) of the method poly()?
2. (2 points) Replace the poly() method with one that has a faster runtime.

10. (3 points) Write the following method which takes in a Set of Strings as input and removes any string which contains a period (‘.’) or a comma (‘,`). Note that this method does not return another Set - it must modify the given Set.

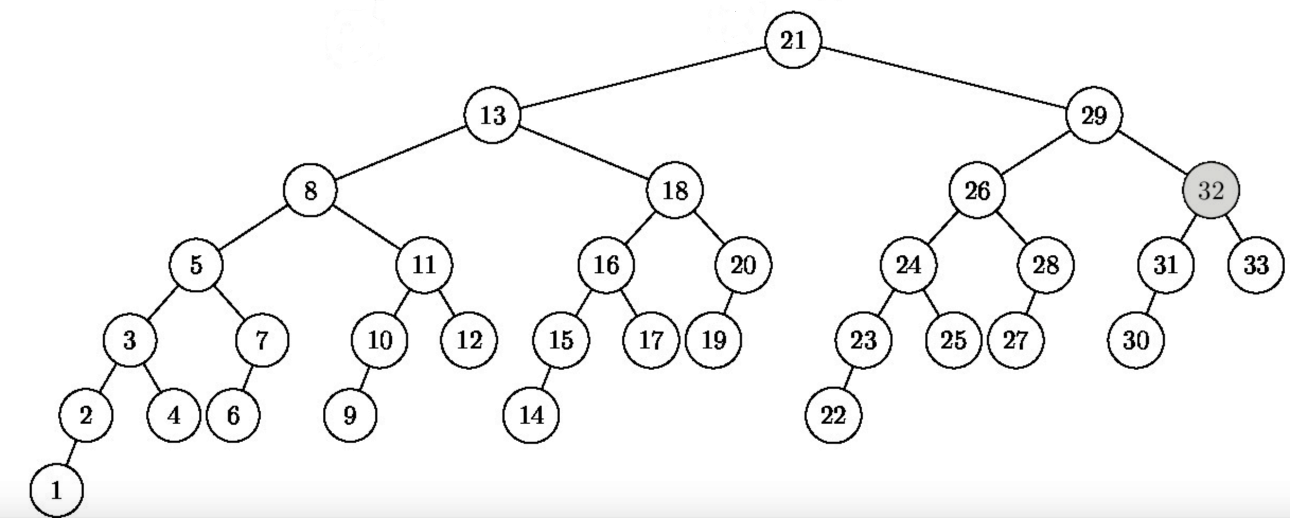
public static void removeSomeStrings(Set<String> s) {

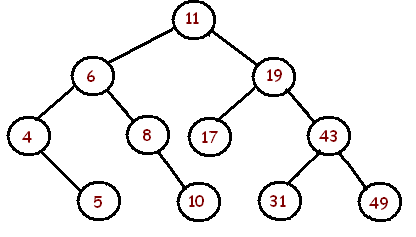
}

1. The runtime of ArrayList’s .add() method is said to be O(1) time, even though it sometimes performs a size-doubling of its underlying array.
2. (2 points) what would the big-O runtime of ArrayList’s .add() method be if the resizing step changed from doubling the array size to increasing the array size by multiplying the size by 5? Justify your answer with 1 or 2 sentences.
3. (2 points) what would the big-O runtime of ArrayList’s .add() method be if the resizing step changed from doubling the array size to increasing the array size by adding 1000 more spaces? Justify your answer with 1 or 2 sentences.
4. (3 points) The runtime of Java’s HashSet operations .add() .contains() and .remove() are described as being O(1) time, but we also learned that Java’s HashSet uses separate chaining which might require some searching work. Explain why the HashSet operations are said to be O(1) time. Use 2 to 4 sentences to answer this question.
5. (3 points) Draw the resulting MAX-heap after inserting the following keys in the order given. Note that a MAX-heap keeps the largest key at the root of the heap.

1, 10, 2, 9, 3, 8, 4, 7, 5, 6

1. Consider the following binary tree:

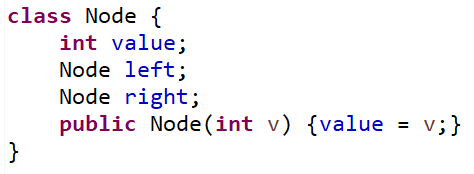
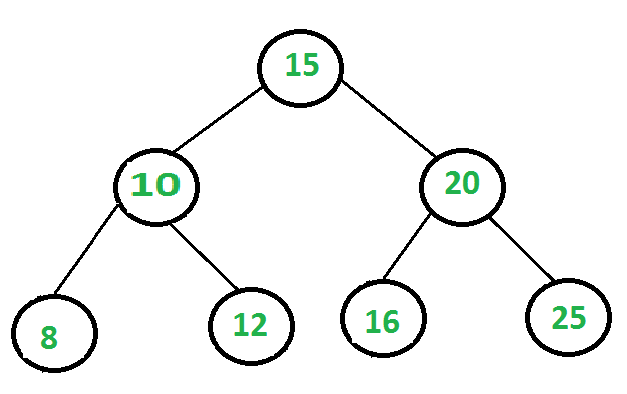


1. (1 point) Is this a valid binary search tree?
2. (2 points) Is this a valid AVL tree? Explain with one sentence.
3. Consider the following AVL tree:  
   
4. (2 points) Show the resulting AVL tree after the operation .add(44)
5. (2 points) Show the resulting AVL tree after the operation .remove(6)
6. (3 points) For the original tree given in question 15 (not your modified trees, but the original one in the question), interpret it as a splay tree and perform the operation .delete(10). Show the resulting splay tree.
7. Imagine hosting a cloud service which is a collection/database of user-submitted photos, sorted by timestamp. You expect millions of people to use your realtime service on one day during a special event, where everyone in attendance at the event are taking photos and submitting them, but many people who cannot make it to the event want to access the photos as well. Because you want fast .add() and .remove() and you also support queries like “retrieve all photos from 2:45pm to 2:47pm,” you decide to use a SkipList since it will have fast .add()s and .contains()s (or retrievals), and to also maintain sorted order.

After the day is done, users will not be submitting any more photos, but people will still want to view photos. Also, you are afraid that the SkipList uses a lot of space on the cloud and you are being charged for that space usage. You consider migrating your online collection to a different data structure.

(4 points) Would you rather move all the photos into a sorted array? Or remove all the extra layers and just keep a sorted linked list? Or would you move everything to a TreeSet? Or maybe a HashSet? Or maybe leaving it as a SkipList is still in your best interests. Which one do you choose and why? Use at most 4 sentences to answer this.

1. (3 points) Given the following Node object and Binary Tree:

write a method that accepts Node root as parameter and produces an output of all the tree nodes which, for the above tree example, would produce the ordering:

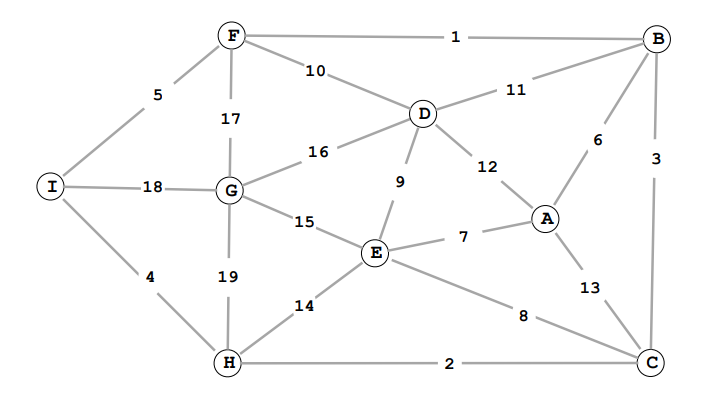
**8 12 10 16 25 20 15**

The method header is given to you. Use System.out.print(Node.value+” “) to write the value of a Node.

public static void output(Node root) {

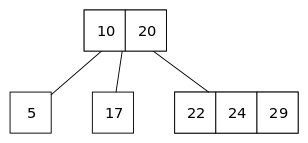
}

1. (4 points) For the following graph, indicate the sequence of edges added to a solution to the Minimum Spanning Tree problem in the order that Kruskal’s algorithm would include them. Do not include any edges which are not part of the minimum spanning tree solution.



Your answer should be a list of edge values, starting with 1.

1. (3 points) Given the following (2,4)-tree:

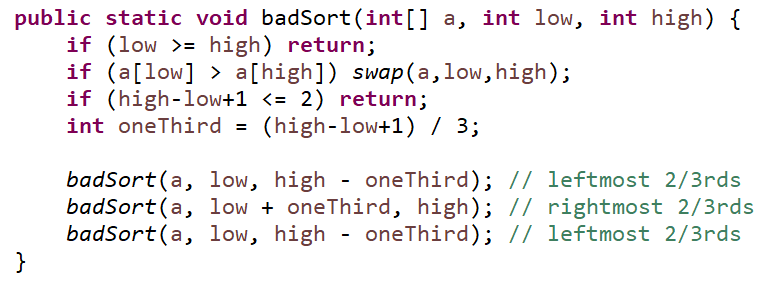


show the resulting (2,4)-tree after adding the following keys in this order: 27, 26, 25.

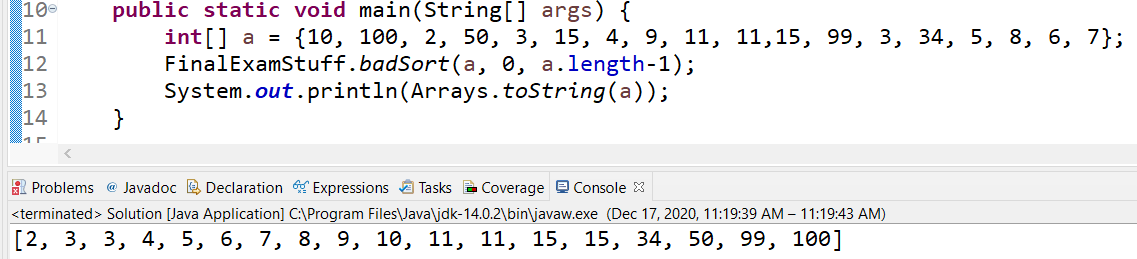
1. (3 points) Find the optimal prefix encoding for the following letters, given their frequency of occurrence

|  |  |
| --- | --- |
| A | 9 |
| T | 15 |
| C | 38 |
| G | 17 |
| U | 7 |

1. Your friend offers you an original sorting method called “badSort”. It does the following: If the first element is larger than the last element, it swaps them. And then it recursively performs this same process on the first two-thirds of the array, then again on the last two-thirds of the array, and then once again on the first two-thirds of the array. Each of the recursive calls make additional recursive calls until a base case of where the subarray considered has less than 3 elements. Here is a Java implementation of this badSort:



Here is a sample output for this method on one example.



1. (2 points) Explain how you would test if your friend’s sorting method is correct. Use 2 to 3 sentences.

.

1. (2 points) Your friend claims that this is better than something like mergeSort because there isn’t a linear amount of work done in each recursive call (like there is no zipper or merge process). Write a recurrence for T(n) where T(n) is the runtime of running badSort on an array of n elements.

1. (1 point) Find a runtime for your recurrence by solving it with any method.. you may plug it into <https://www.wolframalpha.com/> if you prefer.  
   Is the result better than, worse than, or equal to O(n log n) ?
2. (1 point) Assuming your friend’s badSort works as a correct sorting process, is this an in-place sort?
3. (1 point) Assuming your friend’s badSort works as a correct sorting process, is this a stable sort?