

Gradescope Basics

Why Gradescope?

"Gradescope helps you seamlessly administer and grade all of your assessments, whether online or in-class. Save time grading and get a clear picture of how your students are doing."

Grading Submissions

1. Upload submissions (either student or staff)
2. Create rubric (can also create rubric while grading)
 - a. Supports positive scoring (see below), or negative scoring

Final In Person

- ✓ Edit Outline
- ✓ Create Rubric
- ✓ Manage Scans
- ✓ Manage Submissions
- ✓ Grade Submissions
- ✓ Review Grades
- Ⓢ Regrade Requests
- Ⓢ Statistics
- Ⓢ Settings

Q7.5
100 points

$\Theta(\text{_____})$

Item	Score	Feedback
1	+100.0	Correct ($\Theta(N \log(N))$)
2	+75.0	Correct but unsimplified
3	+0.0	Incorrect
4	+0.0	Blank

+ Add Rubric Item + Create Group + Import...

3. (Optional): Group student responses using AI

Grading Method

Grade Individually

Form Answer Groups First

AI Assists With Recognizing

- ☐ Blank answers only (manual grouping)
- ☐ Multiple choice marks
- ☐ Math fill-in-the-blank
- ☒ Text fill-in-the-blank

Track and respond to regrade requests

Previous Requests

Student Request

i and f are technically in correct relative order at some point, and i and f are both present in the last 3 blanks- would provide same results as having i in blank 3 and f in blank 4 for instance

Staff Response (edit)

It would not; for example, if a was false, i would never be checked.

What does Gradescope support?

- Written exams/quizzes/problem sets
- Online, built-in assignments with auto-grading

Q3.1 Operation 1
20 Points

Convert this 2-3 tree to an LLRB, and describe the 6 LLRB operations to balance the tree after inserting the number 11. The LLRB operations are: `rotateRight(x)`, `rotateLeft(x)`, and `colorFlip(x)`. [You may find the lecture on LLRBs helpful.](#)

Operation 1

`rotateLeft(10)`

Explanation

To convert the 2-3 tree, for any 3 nodes (nodes with 2 elements and 3 children), make all smaller elements the left child of the larger element, and connect them with a red link. Make sure that your tree still contains the binary search invariant, which states that all elements to the left of a node are smaller, and all

- Bubble sheets
- Programming assignments

4. Grade student submissions

Total Points
250.0 / 800.0 pts

Rubric Settings
Collapse View

Item	Score	Feedback
1	+150.0	Assigns <code>valid=true</code>
2	+150.0	Correctly assigns <code>index = charToInt(word.charAt(0))</code>
3	-50.0	Minor Error when assigning <code>index</code> : Used <code>intToChar</code> , forgot to prepend <code>word.</code> in <code>charAt</code> , used <code>[]</code> , off by one, etc.
4	--	Null Case No rubric item applied
5	+150.0	Non Null Case +100.0 Called <code>.insert</code> to <code>children[index]</code> +100.0 Used <code>word.substring(1)</code> as argument of <code>insert</code> -50.0 Minor Error: Incorrect java syntax, forgot <code>word</code> in <code>substring</code> , incorrect index, etc.
6	+0.0	Blank/Incorrect

+ Add Rubric Item + Create Group + Import...

Submission Specific Adjustments

Point Adjustment 0

Provide Comments Specific To This Submission

Apply Previously Used Comments

Tip: Use your keyboard!

Make manual adjustments

Leave feedback!

Press "Z" for the next submission