Grading form assignment 2 - Algorithms & Data Structures

Group number: 181

Programming

Try running the **public** unit tests and verify that all of them are passed through a **correct implementation** (that is, make sure that the answers are not hard-coded). If a public unit test does not pass, or the implementation is not what it should be (for example if they were supposed to program a recursive algorithm but did something else), the assignment will not be graded, and a retake can be performed to obtain a grade.

Correctness

- ☑ (3 points) The *public unit* tests are working through a good implementation (and not by cheating the unit tests by e.g., hard-coding answers).

Number of failed private unit tests:	Points for the private unit tests (1-num_failed/6)
0	1

Additional feedback (if any, which private unit tests failed?)

If it is described in the assignment that numbers are sorted, you can assume that the numbers are sorted, you don't have to check that anymore. Also, you don't have to fill the grid yourself, this is done in the unit tests.

Efficiency and coding style

	Feedback	**Points given**	**Max number of points reachable**
Variable names (informative, lowercase: not CamelCase but instead camel_case). Subtract 0.15 per instance of different conventions.	Don't use i and j, they are not descriptive. (-0.15)	0.15	0.3
Efficiency of solutions (not more expensive or complicated than required). Subtract 0.25 per inefficiency. If the divide	Lines 191 and 194 are incorrect, as a value in top right can be smaller than any value on the diagonal except (0,0) (-0.25)	0.75	1

and conquer searches the same region multiple times, subtract 0.5. If the divconq approach computes the middle element, but does not check whether this middle element is equal to the searched value, subtract 0.5.			
Useful comments in functions to explain what is happening. Subtract 0.15 per instance where comments should have been placed.	Don't give every line a comment if it is trivial.	0.2	0.2

Report

Introduction:

- ☑ (1 point) It is clearly explained how the implemented approach works step-by-step: the approach can be re-implemented by just reading the introduction. If the described approach is not the approach implemented by the students, no points can be given here.
- ☑ (0.5 point) A correct explanation is given as to why the proposed approach is more efficient than a simple brute force scan over all cells of the matrix

Additional feedback

Search tree and optimality

- ☑ (0.5 points) The search tree that was drawn is clear, and correct.
- ☐ (0.5 points) It is explained correctly why the approach is optimal: only cells in which the value we are searching for can never occur are discarded

Additional feedback

Greedy approach

- ☐ (0.25 points) A greedy approach is proposed, making locally optimal and irreversible decisions that are feasible to compute
- ☐ (0.25) An adversarial test case is given where the greedy approach does not find the value that we are searching for (while it is in the matrix) OR if an optimal approach is actually given, it is explained why the approach is optimal

Additional feedback

A new greedy approach was asked.

Experiment

- ☑ (0.25 points) The chosen experimental setup is logical for comparing the complexity of the DQ approach and the naive brute force approach, that is, the value that is searched for should not occur in the original rectangle so that the maximum number of cells are searched (or it should be enforced that the searched value is the last to be explored).
- ☑ (0.25) The chosen visualization method is appropriate for analyzing and interpreting the results
- ☑ (0.25) A sufficiently large number of grid sizes have been experimented with and this has been visualized in a proper way (e.g., confine the grids to be squares of n x n and vary n on the x-axis)
- ☐ (0.25) A logical interpretation and analysis of the results is given

Additional feedback

No true analysis is performed.

Points for sophisticated design

- ☐ (0.5 points) Several different divide-and-conquer strategies have been considered and the time complexities of the different approaches have been analyzed. The students implemented either the most efficient one, or experimented with multiple approaches.

Completeness

- ☐ (-0.5 points) The students did *not* hand in a file with their own unit tests

Page limit

Number of extra pages (num_pages-3): Point subtracted (-1*extra_pages)

1	-1

Final grade: 7.7-1=6.7