Week 9 — Sheet 8

Algorithms and Data Structures

06.11.2023 — Georg Hasebe

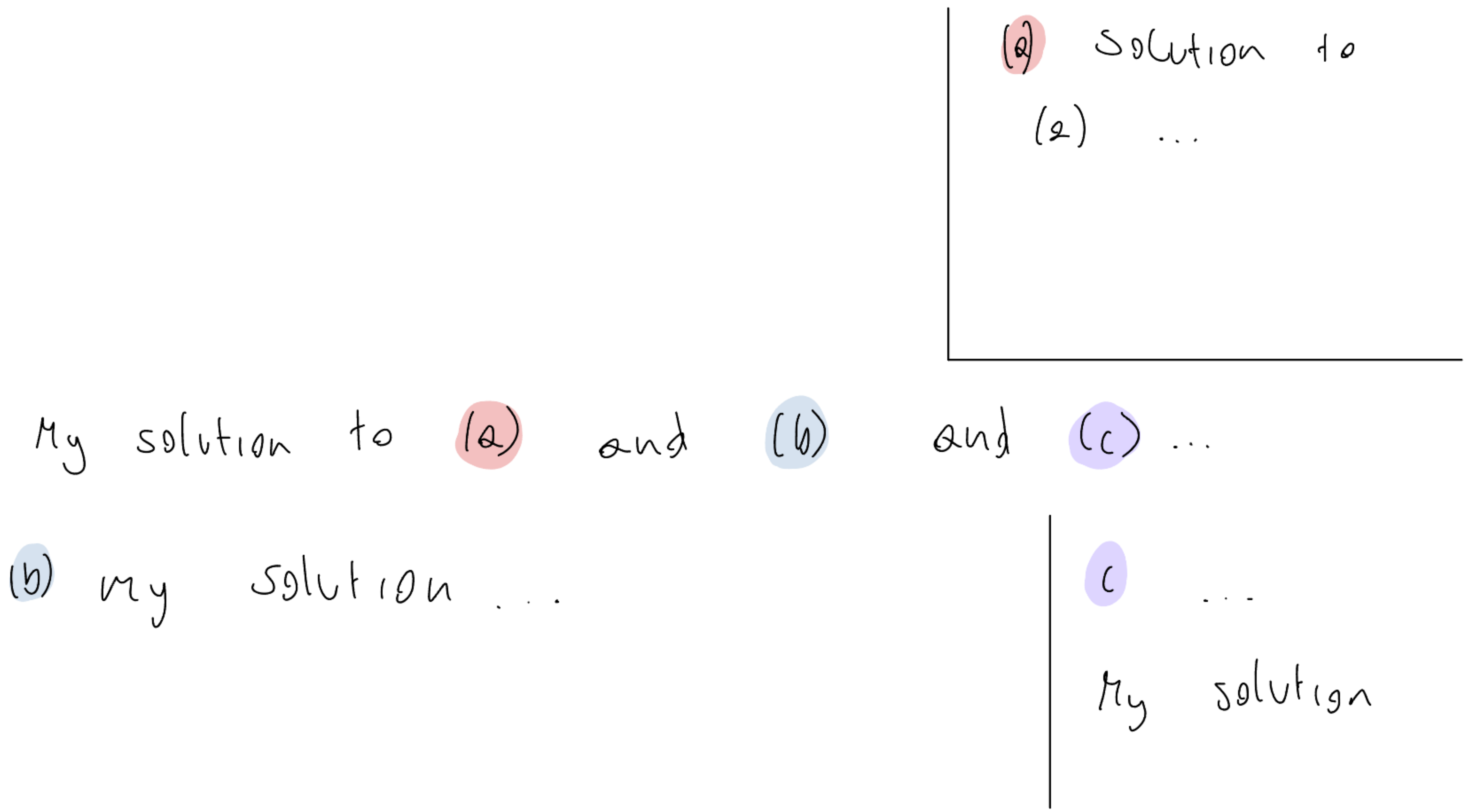
Debriefing of Submissions

Please don't just annotate

In your solution, address the following aspects:

1. *Dimensions of the DP table*: What are the dimensions of the DP table? $n \times n$
2. *Subproblems*: What is the meaning of each entry? dp[i][j] = ...
3. *Recursion*: How can an entry of the table be computed from previous entries? Justify why your recurrence relation is correct. Specify the base cases of the recursion, i.e., the cases that do not depend on others. dp[i][j] = ...
4. *Calculation order*: In which order can entries be computed so that values needed for each entry have been determined in previous steps? from ... to ...
5. *Extracting the solution*: How can the solution be extracted once the table has been filled? dp[n][b]
6. *Running time*: What is the running time of your solution? $O(n \cdot b)$

Jumping around can decrease readability!



Think of the exam!

- Writing on Tablet is very different from paper
- On paper:
  - No dragging stuff around
  - No copy paste
  - No “infinite” space
  - Probably not that many colors (and time to use them)

Structure

To prove: A.

$A \Rightarrow B \Rightarrow C \Rightarrow \ldots \Rightarrow X$ and X is true.

For example: We prove $w(I_k) \leq (1 + \varepsilon) \cdot W$.

$$\Rightarrow \sum_{i \in I_k} w_i \leq \left(1 + \frac{n \cdot k}{w_{\text{max}}}\right) \cdot W \ldots$$

Exercise Sheet 8

Debriefing of Exercise Sheet 8

Dynamic Programming

(Other file)