

# Data Structures and Algorithms

## Laboratory organization

### 1. Laboratory Schedule

- Laboratory is structured as 2-hour classes every two weeks.
- Lab assignments received at a given laboratory have to be turned in at the next laboratory.

Lab	Lab Topic	Labs that can be presented
Lab1	FixedCapBiMap Example	-
Lab2	L1. Dynamic array	-
Lab 3	L2. Linked list with dynamic allocation	<b>L1</b> (regular deadline)
Lab 4	L3. Linked list on array	<b>L2</b> (regular deadline) AND <b>L1</b> (with penalty)
Lab 5	L4. Hash table	<b>L3</b> (regular deadline) AND <b>L2</b> (with penalty)
Lab 6	L5. Binary search tree	<b>L4</b> (regular deadline) OR <b>L3</b> (with penalty)
Lab 7	Presenting lab L5	<b>L5</b> (regular deadline) OR <b>L4</b> (with penalty)

### 2. Laboratory activity and grading

Each of the laboratories L1-L5 focuses on a data structure. Students will receive a container (ADT) and will have to realize an application in C++ to implement the given **container** using a given representation and the given **data structure**, starting from a provided interface of the container.

#### 2.1 Requirements:

- The interface of every ADT (together with the description of the operations) has to be respected (they can be found in the Interfaces folder on MS Teams).
- **It is not allowed to implement a separate class for the data structure. The data structure will be used directly for implementing the given container.**
- Elements of the container will be of the generic type **TElem** (or **TComp** for sorted containers). This is visible in the interface which assumes these types of elements. Nevertheless, for testing the implementation the **TElem = int** (and **TComp = int**) definition will be used.
- Every implemented operation will be followed (or preceded) in the .cpp file by a line of comment, containing the best, worst and total complexity of the algorithm.
- The archives containing the interface of every ADT contain files for testing the implementation as well. When the laboratory is delivered, the application will be tested by the student (in front of the lab teacher) on the 2 test sets (**ShortTest** and **ExtendedTest**) provided for the container.

#### 2.2 Lab delivery process:

- Every student who wants to deliver an assignment, will make sure that the implementation passes the provided test sets (**ShortTest** and **ExtendedTest**). This should be checked at home (and errors should be corrected before the lab).
- In the first part of the lab (approx. 30 minutes) every student who wants to deliver an assignment:
  - will receive a new functionality and:
    - has to implement and test it (in C++)
    - has to deduce the complexity of the functionality (best case, worst case, total complexity).
- When the allocated time is over, the lab teacher will check the application and the extra requirements presented above.

### 2.3 Lab grading:

- Lab assignments are graded in the following way:
  - 1 point: Start
  - 3 points: Application works correctly for the provided tests (1 point if only for short tests)
  - 3 points: Explanations regarding the implementation
  - 1 point: Complexities (0.4 best-case, 0.4 worst-case and 0.2 total complexity).
  - 2 points: Testing and explanations for the new functionality

**Obs:** For maximum grade, the operations of the container need to be implemented efficiently.

### 2.4 The first two labs:

- To prepare the students for the organization of the lab, a video tutorial was prepared, which is available on MS Teams. This tutorial, made of 8 videos, presents the step-by-step development of a lab assignment for an invented container, the FixedCapBiMap. During the first lab, the lab teacher will provide more details about these videos. Until the second lab, every student will have to watch the videos, follow the instructions from it, and implement the example (ADT FixedCapBiMap implemented on an array).
- In the second lab students will have to show their implementation to the lab teacher and they can discuss any issues/problems that appeared during the implementation. Also, during the second lab an example of an extra operation for the FixedCapBiMap will also be discussed (to simulate the entire lab preparation – lab delivery process).
- In the second lab students will also receive a problem from L1. This problem needs to be solved and presented during the third lab.

## **3. Laboratory rules**

- A plagiarized laboratory assignment will receive a grade of **0**. Any laboratory assignment, where the student cannot answer questions regarding the code (both “what does this piece of code

do?” and “why did you decide to implement it this way?” - type of questions) will be considered plagiarized.

- In case of a delay of one lab (two weeks) the grade for the assignment will be **multiplied by 0.8**. Delays greater than one lab (two weeks) are not accepted.
- During Lab 4 and 5 it is possible to present two lab assignments (the regular one and the previous one), in the next labs only one assignment can be presented (either the one which has the regular deadline at that lab, or the previous one with penalty). For more information see the table at the beginning of this document.
- Laboratory attendance is mandatory for **90% of the labs** (6 out of 7 labs). Students who do not have at least 6 attendances at the laboratory cannot participate at the written exam, neither in the regular, nor in the retake session and they cannot pass this course.
- **At most one laboratory attendance** can be recovered with a different group, but only with the explicit agreement of the lab teacher. In this case the assignment grade is computed according to the points from *Lab grading*. In case of illness, absences will be motivated by the lab teacher, based on a medical certificate. **Medical certificates have to be presented at most one week after the absence**, after that period they will not be accepted. The lab teacher needs to be announced about the reason for being absent and a scan/photo of the medical certificate has to be sent by mail to him/her in order to get the absence motivated.
- **Final laboratory grade, LG**, will be computed as the weighted average of the grades received for the 5 lab assignments. If a lab assignment is not delivered, its grade is 0.

$$LG = \frac{12 * L1 + 12 * L2 + 22 * L3 + 27 * L4 + 27 * L5}{100}$$