# Description and Requirements

Iuliana Bocicor, Arthur Molnar
*arthur@cs.ubbcluj.ro*

Babes-Bolyai University

2021

# Guiding Teachers

- Arthur MOLNAR, Lect. PhD

- Andreea-Diana POP, Lect. PhD

- Gabriel MIRCEA, Lect. PhD

- Alexandra ALBU, PhD. undergraduate

- Andrei MIHAI, PhD. undergraduate

- Diana TRUTA, associate prof.

- Razvan BORZA, associate prof.

- **Lecture**: 2 hours/week

- **Seminar**: 1 hour/week

- **Lab**: 2 hours/week

# Objectives

Description
and
Requirements

Iuliana
Bocicor,
Arthur Molnar

**Course Objectives:**

- Introduction to the C/C++ Programming Languages
- Practice important concepts from the object oriented paradigm

**What you should learn from this course:**

- solve small/medium scale problems using OOP
- construct and use class structures as fundamental, modular building blocks
- understand the role of inheritance, polymorphism, dynamic binding and generic structures in building reusable code
- use libraries (STL)
- write small/medium scale C++ programs with simple graphical user interfaces

# Bibliography I

Description
and
Requirements

Iuliana
Bocicor,
Arthur Molnar

1. B. Stroustrup. *The C++ Programming Language*, Addison Wesley, 1998.

2. Bruce Eckel. *Thinking in C++*, Prentice Hall, 1995.

3. A. Alexandrescu. *Programarea moderna in C++: Programare generica si modele de proiectare aplicate*, Editura Teora, 2002.

4. S. Meyers. *Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition)*, Addison-Wesley, 2005.

# Bibliography II

**5** S. Meyers. *More effective C++: 35 New Ways to Improve Your Programs and Designs*, Addison-Wesley, 1995.

**6** B. Stroustrup. *A Tour of C++*, Addison Wesley, 2013.

**7** E. Gamma, R. Helm, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley Longman Publishing, 1995.

**8** C++ reference (http://en.cppreference.com/w/).

**9** Qt Documentation (http://doc.qt.io/qt-5/).

Description
and
Requirements

Iuliana
Bocicor,
Arthur Molnar

# Course Content

Description
and
Requirements

Iuliana
Bocicor,
Arthur Molnar

1. C Programming Language Fundamentals, Functions
2. Modular Programming and Memory Management
3. Classes and Objects
4. Templates and the C++ Standard Library
5. Inheritance and Polymorphism
6. I/O Streams and Exceptions
7. Pointers and RAII
8. Designing GUI with Qt
9. Design Patterns

# Course Rules I

- Attendance is compulsory for laboratory and seminar activities:
    - Laboratory: at least **12** out of a total of 14.
    - Seminar: at least **5** out of a total of 7.

- **Unless you have the required number of attendances, you are not allowed to take the examination this year!**

- Do not plagiarise!

# Course Rules II

- Another precondition to attend the examination in the regular session is to have the laboratory grade **L** $\geq 5$ (no rounding)! This grade is computed as shown on the next slide.

- If you do not have at least 5 for your laboratory activity, you may participate to the examination only during the retake session. In this case, you must present all laboratory assignments and the maximum grade you can get for these is 5.

# Grading

Description
and
Requirements

Iuliana
Bocicor,
Arthur Molnar

- Lab grade (**L**)
    - 50% Lab assignments
    - 50%: 3 practical tests during the labs (10%, 15%, 25%)
    - Bonus work (**LB**)
- Written examination (**W**)
- Practical examination (**P**)
- Seminar activity (**SA**): 0 - 0.5 (bonus)

**The final grade is computed as:**

$$G = 0.4 \cdot L + 0.3 \cdot P + 0.3 \cdot W + SA + LB$$

**To pass the course, all grades (L, W, P) must be $\geq 5$ (no rounding).**