Exercise session 02

Introduction to C++.

Advanced Programming - SISSA, UniTS, 2025-2026

Giuseppe Alessio D'Inverno

08 Oct 2025

Exercise 1: control structures

Write a C++ program temperature_converter that converts a given temperature from Celsius to Fahrenheit or viceversa.

- The program takes an input number from the user and the input unit as a string (or character).
- Use conditional statements to check if the user has provided a temperature in Fahrenheit or Celsius and print the corresponding output.
- The conversion formulas are:

$$T_{
m Fahrenheit} = rac{9}{5} T_{
m Celsius} + 32. \ T_{
m Celsius} = rac{5}{9} (T_{
m Fahrenheit} - 32).$$

Exercise 2: memory management

Create a C++ program that dynamically allocates memory for an array of integers.

- Allow the user to specify the size of the array.
- Fill the array with random integers.
- Write a function to find and display the maximum and minimum values in the array. **Hint**: the maximum and minimum value are stored in two variables passed as references to this function.

Exercise 3: complete the missing statistics calculator

The hints/ex3/ folder provides a partially implemented C++ program for calculating and displaying statistics for a set of numbers. Your task is to fill in the missing parts to make the program functional.

- Use the provided function prototypes in statistics.hpp to guide your implementation in statistics.cpp .
- You may need to sort the input numbers to calculate the median. You can use standard C++ functions like std::sort for this purpose.
- Implement a loop in main.cpp to allow the user to calculate statistics for multiple sets of numbers without exiting the program.

Exercise 4: struct

- Define a struct called Student that represents information about a student, including their name, age, and grade average.
- Create a std::vector of 5 Student objects and initialize them with sample data.
- Write a function to display the information of all students in the array.
- Organize your code by separating the struct definition, data initialization, and display function into different files or modules.

Exercise 5: code organization

Write a C++ program that simulates a simple calculator. Define functions for addition, subtraction, multiplication, and division. Allow the user to enter two numbers and choose an operation. Perform the chosen operation and display the result.

Here's a breakdown of the project structure:

- calculator/
 - o src/
 - main.cpp
 - calculator.cpp
 - o include/
 - calculator.hpp
 - build/ (build artifacts, such as object files and executables).
 - build.sh (a Bash script that compiles and properly links the code)