Advanced Methods for Scientific Computing (AMSC)
Lecture title: Catch2: A Modern C++ Unit Testing Framework

Luca Formaggia

MOX
Dipartimento di Matematica
Politecnico di Milano

A.Y. 2024/2025

Introduction to Catch2

- Catch2 is a modern C++-native test framework for unit-tests, TDD (Test-Driven Development), and BDD (Behavior-Driven Development).
- Supports C++14, C++17, and later standards.
- Provides basic micro-benchmarking features and simple BDD macros.

Key Features

- Simple and natural syntax for writing tests.
- Test names do not have to be valid identifiers.
- Assertions look like normal C++ boolean expressions.
- Sections provide a local way to share set-up and tear-down code in tests.

Example Unit Test

```
#include <catch2/catch_test_macros.hpp>
#include <cstdint>

uint32_t factorial(uint32_t number) {
    return number <= 1 ? number : factorial(number - 1) * number;
}

TEST_CASE("Factorials are computed", "[factorial]") {
    REQUIRE(factorial(1) == 1);
    REQUIRE(factorial(2) == 2);
    REQUIRE(factorial(3) == 6);
    REQUIRE(factorial(10) == 3628800);
}
```

Example Microbenchmark

```cpp
#include <catch2/catch_test_macros.hpp>
#include <catch2/benchmark/catch_benchmark.hpp>
#include <cstdint>

uint64_t fibonacci(uint64_t number) {
    return number<2 ?number:fibonacci(number-1) + fibonacci(number-2);
}

TEST_CASE("Benchmark Fibonacci", "[!benchmark]") {
    REQUIRE(fibonacci(5) == 5);
    REQUIRE(fibonacci(20) == 6765);
    BENCHMARK("fibonacci 20") {
        return fibonacci(20);
    };
    REQUIRE(fibonacci(25) == 75025);
    BENCHMARK("fibonacci 25") {
        return fibonacci(25);
    };
}
```

Getting Started with Catch2

- The documentation includes a tutorial for getting started.
- Provides a reference section with detailed information.
- Migration guidelines from v2 to v3 are available.

Further Resources

- GitHub Repository:
  `https://github.com/catchorg/Catch2`
- For discussions or questions, join the Catch2 Discord community.