UNIVERSITÉ DU LUXEMBOURG

Vector Data Structure

Parallel Computing

Goals

- ✯ Learn C++ class and object.
- ✯ Learn C++ template class.
- ✯ Implement a dynamic array data structure.
- ✯ **Relevant videos**:
  - Pointer type
  - Class type
  - Memory allocation
  - Operator overloading
  - Template (from exercise 2)
  - Move semantics (from exercise 2)

Deliverables

1. The code on your Github repository generated by clicking here: `https://classroom.github.com/a/yLVUU_0U`
2. **Reviewer:** Wei Huang (Github: `multivvac`)

Exercise 1 – Vector of integers

Implement the necessary constructors, destructors and methods given in the file `int_vector.hpp`. For additionnal documentation see std::vector, we try to follow it as close as possible.

Exercise 2 – Templated vector class

1. Update the previous example using a templated class in the file `vector.hpp`.
2. Add a move constructor.

Exercise 3 – Placement new

Allow the template type `T` to lack a default constructor, for instance:

```
struct S {
  S(int x) {}
};

Vector<S> x;
x.reserve(10);
```

See placement new and std::malloc.