Exercise

For all rooms

(1) Create files `biginteger.h` and `biginteger.cpp` to declare and implement a `class` called **`BigInteger`**

The only (private) attribute of this `class` will be a `std::vector<char> _v` for storing digits.

For example, _v = [‘9’,‘9’,‘9’,‘9’,‘9’,‘9’] represents the number 999 999.

(2) Overload operator + for sum between two **`BigInteger`** objects. (Optional: also overload +=)

(3) Overload operator >> for reading a **`BigInteger`** object.

(4) Overload operator << for printing a **`BigInteger`** object.

(5) Overload operator < for comparing two **`BigInteger`** objects.

(6) Test your class as follows, in a `main.cpp` file:

- Read 10 **`BigInteger`** objects (with >>) and put them in a vector `std::vector<BigInteger> bigs;`

- Sort the vector `bigs` calling `std::sort` function from `#include<algorithm>`

- Print all elements inside the vector `bigs` (with <<)

- Sum all the elements of the vector `bigs` and print the result.