

Object Serialization

Object Oriented Programming Project in ZJU.

Compile and Run

Using commands in terminal to construct:

```
1 | mkdir build
2 | cd build
3 | cmake ..
4 | make
5 | cd ..
```

To make things easy, I provided a shell **test.sh**. We can simply use the command `./test.sh` to construct.

After that, using commands in terminal to run:

```
1 | mkdir test
2 | rm -f test/*
3 | cd bin
4 | ./test.out
```

Test





All test results will be generated in directory `./bin/test_all.txt`. Some Intermediate results will be generated in directory `./test/*`.

Note: Intermediate results of XML serialization are deleted in the process of XML deserialization. If you want to see the intermediate results, you can comment the part of XML deserialization.

Using commands `./bin/test.out` to check whether all object can be serialization / deserialization correctly.

<pre>[=====] Running 2 tests from 1 test suite [-----] Global test environment set-up [RUN] test.BinarySerializationTest [OK] BinarySerializationTest [RUN] test.XMLSerializationTest [FAILED] XMLSerializationTest [-----] Global test environment tear-down [FAILED] 1 errors.</pre>	<pre>[=====] Running 2 tests from 1 test suite [-----] Global test environment set-up [RUN] test.BinarySerializationTest [OK] BinarySerializationTest [RUN] test.XMLSerializationTest [OK] BinarySerializationTest [-----] Global test environment tear-down [PASSED] 2 tests.</pre>
---	---

Functions

- ☑️  Supported binary serialization/deserialization from a std type and user defined type
- ☑️  Supported binary serialization/deserialization from smart pointers
- ☑️  Supported XML serialization/deserialization from a std type and user defined type
- ☑️  Supported XML serialization/deserialization from smart pointers

Bugs

Due to some bugs in `tinyxml2`, so there might be some bugs in XML serialization.

Address

Project address : [My Github](#)

@copyright Copyright (c) 2022