

Lesson 3:
Advanced OOP

SEARCH

RESOURCES

CONCEPTS

1. Polymorphism and inheritance

2. Bjarne on Inheritance

3. Inheritance

4. Access Specifiers

5. Exercise: Animal Class

6. Composition

7. Exercise: Class Hierarchy

8. Exercise: Friends

9. Polymorphism: Overloading

10. Polymorphism: Operator Overlo...

11. Virtual Functions

12. Polymorphism: Overriding

13. Override

14. Multiple Inheritance

15. Generic Programming

16. Bjarne on Generic Programming

17. Templates

18. Bjarne on Templates

19. Exercise: Comparison Operation

20. Deduction

21. Exercise: Class Template

22. Summary

23. Bjarne on Best Practices with Cla...

Exercise: Class Template

SEND FEEDBACK

In 1 1:

#include <assert.h>
#include <string>
#include <sstream>

// TODO: Add the correct template specification
template<typename KeyType, typename ValueType>
class Mapping {
public:

 Mapping(KeyType key, ValueType value) : key(key), value(value) {}
 std::string Print() const {
 std::ostringstream stream;
 stream << key << " : " << value;
 return stream.str();
 }
 KeyType key;
 ValueType value;
};

// Test
int main() {
 Mapping<std::string, int> mapping("age", 20);
 assert(mapping.Print() == "age: 20");
}

Compile & Run

Explain

Loading terminal (id_glvv5), please wait...

Loading [MathJax]/extensions/Safe.js

Menu

Shrink

NEXT