Task 9

Error Handling

- What are the keywords of error handling in C++? What things can be throw and catch?
- Give 4 types of `std::exception` in `std` library.
- What the program will do if we ignore the thrown exceptions?
- How to define a custom exception? Give an example.
- Give an example of error handling. (You can use your project.)
- What is the output of the following program? Try to explain why.

```cpp
#include <iostream>
#include <stdexcept>
void f(bool b) {
    if (b) throw std::runtime_error("Here is a runtime error.");
    throw std::logic_error("Here is a logic error.");
}
int main() {
    try { f(true);  }
    catch (std::exception e) {
        std::cout << e.what() << std::endl;
    }
    try { f(true); }
    catch (std::exception &e) {
        std::cout << e.what() << std::endl;
    }
    try { f(false); }
    catch (std::logic_error e) {
        std::cout << e.what() << std::endl;
    } catch (std::exception &e) {
        std::cout << e.what() << std::endl;
    }
}
```

- What will the following program do with different `value`? Should we throw exceptions in destructor and constructors? If you can't successfully compile it, you can try it on Compiler Explorer.

```cpp
#include <iostream>
#include <stdexcept>
struct Inner {
    Inner(int x) {
        if (x > 1000) throw std::out_of_range("A out of range error.");
    }
};
class A {
public:
    A(int x) { ptr = new Inner(x); }
    ~A() { throw std::logic_error("A logic error."); }
private:
    Inner *ptr;
};
```

```
int main() {
    int value = 10;
    try { A a{value}; }
    catch (std::exception &e) { std::cout << e.what() << std::endl; }
}
```

- What problem does the following program have? How to fix it?

```
#include <iostream>
#include <stdexcept>
struct Inner {
    Inner(int x) {
        if (x > 1000) throw std::out_of_range("Out of range!");
    }
    ~Inner() { std::cout << "Destructor!" << std::endl; }
};
class A {
public:
    A() {
        p1 = new Inner(10);
        p2 = new Inner(1002);
    }
private:
    Inner *p1, *p2;
};
int main() {
    try { A a{}; }
    catch (std::exception &e) { std::cout << e.what() << std::endl; }
}
```

- What does the keyword `noexcept` mean? When should we use it? What is the advantage of it?