C++ course – Exercises Set 9

Wouter Verkerke (Jan 2021)

Exercise 9.1 – Exception handling

The goal of this exercise is to incorporate the concept of 'exception handling' in the `Array` *class*

- Insert the provided exception class `ArrayOutOfBounds` into `Array.hh`

```
#include <iostream>
using namespace std;
class ArrayOutOfBounds {
public:
  ArrayOutOfBounds(int index) : _index(index) {} ;
  int index() const { return _index ; }

private:
  int _index ;
} ;

ostream& operator<<(ostream& os,const ArrayOutOfBounds& m) {
  return (os << "Array index is out of bounds");
}
```

- Modify the both const and non-const versions of the `operator[]` function of class `Array` to throw an `ArrayOutOfBounds` object as exception whenever the passed index is outside the range of the array.

  Pass the offending index to the constructor of the exception object, as documentation of the precise error condition

- Use the `Array` class in your main program in a way that triggers the error condition, i.e. access a position of the array that is outside the allocated range. *Explain what happens*

- Modify your `main()` function to catch the `ArrayOutOfBounds` exception.
  In the error handling part print the exception object do document the clarify the error condition to the user.