Seminar 2

Useful links:

- Named requirements:

  https://en.cppreference.com/w/cpp/named_req

- Allocator:

  https://en.cppreference.com/w/cpp/named_req/Allocator

Let's deal with problems of copying allocators we faced on the last lecture.

Vector:

- Copying

  No need for block of code in comment because the vector is already created in the delegated c-tor. In case of throw, it will be destroyed by destroctor

- operator=

  Trial cases

```
class Vector {
    T* buffer;
    size_t cap;
    size_t size;
    Allocator alloc;

    Vector(const Vector& other)
        : Vector(other.alloc.select_on_container_copy_construction()) {
        reserve(other.size());
       // try {
            for (auto& elem : other) {
                push_back(elem);
            }
       // } catch (...) {
       //     while (!empty()) {
       //         pop_back();
       //     }
       //     shrink_to_fit();
       //     throw;
       // }
    }

    Vector& operator=(const Vector& other) {
        Allocator newalloc = other.alloc.propagate_on_container_copy_assignment::value == st
        Vector newvector(newalloc, other.begin(), other.end());
        swap(newvector);
```

```
    }
}
```