Ex 2 – dry

Yotam Freund, Daniel Belokon

```cpp
class BadInput : public std::exception
{
};

template <class T>
std::vector<T> slice(std::vector<T> vec, int start, int step, int stop)
{
    int size = vec.size();
    if (start < 0 || start >= size || stop < 0 || stop > size || step <= 0)
        throw BadInput();

    std::vector<T> sliced_vec;
    for (auto i = vec.begin() + start; i < vec.begin() + stop; i += step)
    {
        sliced_vec.push_back(*i);
    }

    return sliced_vec;
}

// PART 2

class A
{
public:
    std::vector<std::shared_ptr<int>> values;
    void add(int x) { values.push_back(std::make_shared<int>(x)); }
};
```