```cpp
#include "ShakerSortableIntVector.h"
#include "SortableIntVector.h"
#include <cstddef>
ShakerSortableIntVector::ShakerSortableIntVector(const int aArrayOfIntegers []),
                                                 size_t aNumberOfElements)
    : SortableIntVector(aArrayOfIntegers, aNumberOfElements) {}

void ShakerSortableIntVector::sort(Comparable aOrderFunction) {
    size_t n = this->size();
    size_t beginIndex = 0, endIndex = n - 1;
    while (beginIndex < endIndex) {
        for (size_t i = beginIndex; i <= endIndex - 1; i++) {
            // if a[i] > a[i + 1]

            // aOrderFunction means a <= b => inverse of that is a > b
            if (aOrderFunction(this->get(i), this->get(i + 1))) {
                this->swap(i, i + 1);
            }
        }

        endIndex -= 1;

        for (size_t i = endIndex; i >= beginIndex + 1; i--) {
            if (!aOrderFunction(this->get(i), this->get(i - 1)) &&
                this->get(i) != this->get(i - 1)) {

                this->swap(i, i - 1);
            }
        }

        beginIndex += 1;
    }
}
```