```cpp
#include "IntVector.h"
#include <cstddef>
#include "IntSorter.h"
#include "IntVectorIterator.h"
#include <stdexcept>

using namespace std;

IntVector::IntVector(const int aArrayOfIntegers[], size_t aNumberOfElements)
{
    fNumberOfElements = aNumberOfElements;
    fElements = new int[fNumberOfElements];

    for (size_t i = 0; i < fNumberOfElements; i++)
    {
        fElements[i] = aArrayOfIntegers[i];
    }
}

IntVector::~IntVector()
{
    delete[] fElements;
}

size_t IntVector::size() const
{
    return fNumberOfElements;
}

void IntVector::swap(size_t aSourceIndex, size_t aTargetIndex)
{
    if (aSourceIndex < fNumberOfElements && aTargetIndex < fNumberOfElements)
    {
        size_t tempIndex = fElements[aSourceIndex];
        fElements[aSourceIndex] = fElements[aTargetIndex];
        fElements[aTargetIndex] = tempIndex;

        return;
    }
    throw
        out_of_range("Invalid index(es).");
}

void IntVector::sort(const IntSorter& aSorter)
{
    aSorter(*this);
}

const int IntVector::operator[](size_t aIndex) const
{
    if (aIndex < fNumberOfElements)
    {
        return fElements[aIndex];
```