

Задача 2(Swap, SortPointers, ReversedCopy)

В этой задаче вам нужно реализовать три шаблонных функции:

- `template <typename T> void Swap(T* first, T* second)` — обменивает местами значения, на которые указывают указатели `first` и `second`
- `template <typename T> void SortPointers(vector<T*>& pointers)` — сортирует указатели по значениям, на которые они указывают
- `template <typename T> void ReversedCopy(T* src, size_t count, T* dst)` — копирует в обратном порядке `count` элементов, начиная с адреса в указателе `src`, в область памяти, начинающуюся по адресу `dst`. При этом
 1. каждый объект из диапазона `[src; src + count)` должен быть скопирован не более одного раза;
 2. диапазоны `[src; src + count)` и `[dst; dst + count)` могут пересекаться;
 3. элементы в части диапазона `[src; src + count)`, которая не пересекается с `[dst; dst + count)`, должны остаться неизменными.

Примеры применения шаблонов смотрите в тестах в заготовке решения.

Файл: swap_sort_copy.cpp