

Протокол тестирования:

1. Среда тестирования: Программный продукт написан на C # и будет протестирован в операционной системе Windows. Тестовая среда должна включать:
 - тестовый компьютер с ос Windows;
 - установлена Visual Studio;
 - примеры наборов данных для сортировки.
2. Тестовые примеры: для проверки программного продукта будут выполнены следующие тестовые примеры:
 - а) Тестовый пример 1 - Алгоритмы сортировки:
 - Входные данные: набор данных, содержащий целые числа
 - Результат: отсортированный набор данных
 - Этапы:
 1. загрузите программный продукт;
 2. введите набор данных;
 3. выберите алгоритм сортировки (сортировка по оболочке, пузырьковая сортировка, сортировка по вставке или сортировка по выбору);
 4. выполните алгоритм сортировки;
 5. убедитесь, что вывод представляет собой отсортированный набор данных;
 6. Повторите шаги 3-5 для каждого алгоритма сортировки.
 - б) Тестовый пример 2 - Время обработки запроса:
 - Входные данные: набор данных, содержащий целые числа
 - Результат: время обработки для каждого алгоритма сортировки
 - Этапы:
 1. загрузите программный продукт;
 2. введите набор данных;

3. выберите алгоритм сортировки (сортировка по оболочке, пузырьковая сортировка, сортировка по вставке или сортировка по выбору);
4. выполните алгоритм сортировки;
5. запись времени обработки;
6. повторите шаги 3-5 для каждого алгоритма сортировки;
7. убедитесь, что время обработки отображается в графическом интерфейсе для каждого алгоритма.

с) Тестовый пример 3 - Графический пользовательский интерфейс:

- Входные данные: набор данных, содержащий целые числа
- Результат: отсортированный набор данных и время обработки
- Этапы:
 1. загрузите программный продукт;
 2. введите набор данных;
 3. выберите алгоритм сортировки (сортировка по оболочке, пузырьковая сортировка, сортировка по вставке или сортировка по выбору);
 4. выполните алгоритм сортировки;
 5. убедитесь, что вывод представляет собой отсортированный набор данных;
 6. запись времени обработки;
 7. убедитесь, что время обработки отображается в графическом интерфейсе;
 8. убедитесь, что отсортированный набор данных отображается в графическом интерфейсе;

3. Риски тестирования:

- программный продукт, может быть, не в состоянии обрабатывать большие наборы данных;
- алгоритмы сортировки могут неправильно работать с определенными типами данных;

- графический интерфейс может неточно отображать результаты.

4. Критерии приемлемости:

- все тестовые примеры проходят;
- программный продукт соответствует требованиям, указанным в объеме проекта;
- программный продукт является надежным, эффективным и удобным в использовании.