

ТЕСТИРОВАНИЕ

Тест-требования

К разработанному приложению были представлены следующие тест-требования:

1 Протестировать ввод количества элементов массива.

1.1 Проверить, что при вводе некорректных данных: букв, пустого значения, отрицательного значения, программа оповестит об этом и попросит повторить ввод, только уже с корректным значением.

1.2 Проверить, что при вводе целого положительного значения программа продолжит свою работу и создаст массив.

2 Протестировать заполнение массива.

2.1 Проверить, что при вводе корректных данных программа продолжит работать и запишет все данные в массив.

2.2 Проверить, что при вводе некорректных данных: для всех пустого значения, в некоторых случаях букв и отрицательного значения, программа попросит повторно ввести значение или же оповестит об ошибке ввода и попросит ввести значение повторно.

3 Протестировать запись данных в файл.

Проверить, что при вводе корректных данных программа запишет их в текстовый файл.

4 Протестировать сортировку массива.

Проверить, что при вводе всех корректных данных и при их записи в текстовый файл, данные отобразятся в отсортированном виде.

Тест-план

На основе данных тест-требований был разработан тест-план, состоящий из 4 тестов.

Тестовый пример 1.

Требования: 1.

Описание теста: протестировать функцию ввода длины массива путем введения корректных и некорректных данных.

Входные данные:

Негативные (1.1): Количество элементов массива: «qwe»; Количество элементов массива:«(пустое значение)»; Количество элементов массива:«-20».

Позитивные (1.2): Количество элементов массива: «2».

Ожидаемые выходные данные:

Негативные (1.1): Вывод сообщения об ошибке, повторный ввод значения.

Позитивные (1.2): Продолжение работы программы.

Сценарий негативного тестирования:

- 1) Запускаем программу.
- 2) При запросе ввести размер массива ввести «qwe», «(пустое значение)», «-20».

Сценарий позитивного тестирования:

- 1) Запускаем программу.
- 2) При запросе ввести размер массива ввести «2».

Тестовый пример 2.

Требования: 2.

Описание теста: протестировать заполнение массива после ввода его длины.

Входные данные:

Позитивные (2.1): Количество элементов массива: «2»; продолжительность: «14 дней», цена: «1500», размер туристической группы:

«10»; продолжительность: «10 дней», цена: «1200», размер туристической группы: «12».

Негативные (2.2): Количество элементов массива: «1», продолжительность: «(пустое значение)», цена: «(пустое значение)», «-200», «сто», размер туристической группы: «(пустое значение)», «-200», «сто».

Ожидаемые выходные данные:

Позитивные (2.1): Продолжение работы программы.

Негативные (2.2): Повторный ввод на пустое значение, сообщение о некорректном вводе при вводе отрицательного числа и букв и повторный ввод.

Сценарий позитивного тестирования:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «2».
- 3) При запросе ввести продолжительность ввести «14 дней».
- 4) При запросе ввести цену ввести «1500».
- 5) При запросе ввести размер туристической группы ввести «10».
- 6) При запросе ввести продолжительность ввести «10 дней».
- 7) При запросе ввести цену ввести «1200».
- 8) При запросе ввести размер туристической группы ввести «12».

Сценарий негативного тестирования:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «1».
- 3) При запросе ввести продолжительность ввести «(пустое значение)».
- 4) При запросе ввести цену ввести «(пустое значение)».
- 5) При запросе ввести цену ввести «-200».
- 6) При запросе ввести цену ввести «сто».
- 7) При запросе ввести размер туристической группы ввести «(пустое значение)».
- 8) При запросе ввести размер туристической группы ввести «-200».
- 9) При запросе ввести размер туристической группы ввести «сто».

Тестовый пример 3.

Требования: 3.

Описание теста: протестировать сохранение данных в файл.

Входные данные: Количество элементов массива: «2»; продолжительность: «14 дней», цена: «1500», размер туристической группы: «10»; продолжительность: «10 дней», цена: «1200», размер туристической группы: «12».

Ожидаемые выходные данные:

Сообщение об успешном сохранении.

Сценарий:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «2».
- 3) При запросе ввести продолжительность ввести «14 дней».
- 4) При запросе ввести цену ввести «1500».
- 5) При запросе ввести размер туристической группы ввести «10».
- 6) При запросе ввести продолжительность ввести «10 дней».
- 7) При запросе ввести цену ввести «1200».
- 8) При запросе ввести размер туристической группы ввести «12».
- 9) Открыть файл и убедиться, что все записалось.

Тестовый пример 4.

Требования: 4.

Описание теста: протестировать сортировку массива.

Входные данные: Количество элементов массива: «2»; продолжительность: «14 дней», цена: «1500», размер туристической группы: «10»; продолжительность: «10 дней», цена: «1200», размер туристической группы: «12».

Ожидаемые выходные данные:

Сообщение об успешной сортировке.

Сценарий:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «2».
- 3) При запросе ввести продолжительность ввести «14 дней».
- 4) При запросе ввести цену ввести «1500».
- 5) При запросе ввести размер туристической группы ввести «10».
- 6) При запросе ввести продолжительность ввести «10 дней».
- 7) При запросе ввести цену ввести «1200».
- 8) При запросе ввести размер туристической группы ввести «12».
- 9) Открыть файл и проверить правильности сортировки.

Результаты тестирования

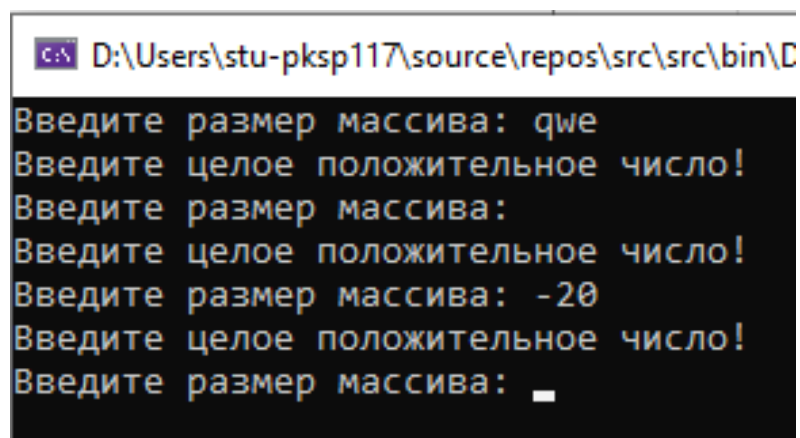
Тестовый пример 1.

Сценарий негативного тестирования:

- 3) Запускаем программу.
- 4) При запросе ввести размер массива ввести «qwe», «(пустое значение)», «-20».

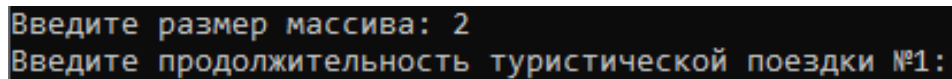
Сценарий позитивного тестирования:

- 3) Запускаем программу.
- 4) При запросе ввести размер массива ввести «2».



```
C:\> D:\Users\stu-pksp117\source\repos\src\src\bin\Debug
Введите размер массива: qwe
Введите целое положительное число!
Введите размер массива:
Введите целое положительное число!
Введите размер массива: -20
Введите целое положительное число!
Введите размер массива: _
```

Рисунок 1 – Результаты негативного тестирования



Введите размер массива: 2
Введите продолжительность туристической поездки №1:

Рисунок 2 – Результат позитивного тестирования

Вывод: результаты соответствуют ожидаемым данным.

Тестовый пример 2.

Сценарий позитивного тестирования:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «2».
- 3) При запросе ввести продолжительность ввести «14 дней».
- 4) При запросе ввести цену ввести «1500».
- 5) При запросе ввести размер туристической группы ввести «10».
- 6) При запросе ввести продолжительность ввести «10 дней».
- 7) При запросе ввести цену ввести «1200».
- 8) При запросе ввести размер туристической группы ввести «12».

Сценарий негативного тестирования:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «1».
- 3) При запросе ввести продолжительность ввести «(пустое значение)»,
затем при повторном «10 дней».
- 4) При запросе ввести цену ввести «(пустое значение)».
- 5) При запросе ввести цену ввести «-200».
- 6) При запросе ввести цену ввести «сто», затем при повторном «1500».
- 7) При запросе ввести размер туристической группы ввести «(пустое
значение)».
- 8) При запросе ввести размер туристической группы ввести «-200».
- 9) При запросе ввести размер туристической группы ввести «сто»,
затем при повторном «10».

```
D:\Users\stu-pksp117\source\repos\src\src\bin\Debug\netcoreapp3.1\src.exe
Введите размер массива: 2
Введите продолжительность туристической поездки №1: 14 дней
Введите цену туристической поездки №1: 1500
Введите размер туристической группы №1: 10
Введите продолжительность туристической поездки №2: 10 дней
Введите цену туристической поездки №2: 1200
Введите размер туристической группы №2: 12
Отсортировано!
Сохранение прошло успешно!
```

Рисунок 3 – Результат позитивного тестирования

```
D:\Users\stu-pksp117\source\repos\src\src\bin\Debug\netcoreapp3.1\src.exe
Введите размер массива: 1
Введите продолжительность туристической поездки №1:
Введите продолжительность туристической поездки №1: 10 дней
Введите цену туристической поездки №1:
Введите цену туристической поездки №1: -200
Введите целое положительное число!
Введите цену туристической поездки №1: сто
Введите целое положительное число!
Введите цену туристической поездки №1: 1500
Введите размер туристической группы №1:
Введите размер туристической группы №1: -200
Введите целое положительное число!
Введите размер туристической группы №1: сто
Введите целое положительное число!
Введите размер туристической группы №1: 10
Отсортировано!
Сохранение прошло успешно!
```

Рисунок 4 – Результат негативного тестирования

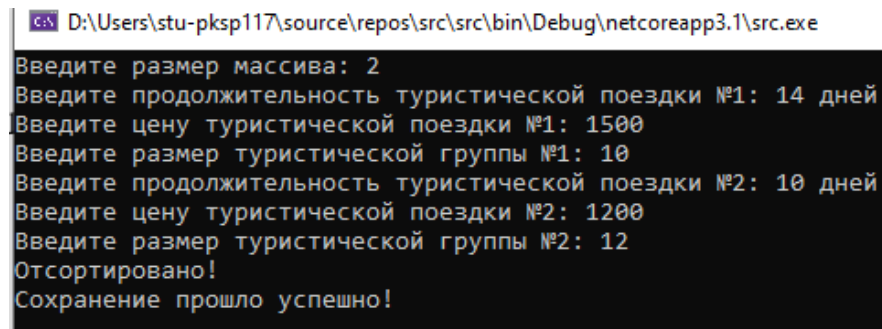
Вывод: результаты соответствуют ожидаемым данным.

Тестовый пример 3.

Сценарий:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «2».
- 3) При запросе ввести продолжительность ввести «14 дней».
- 4) При запросе ввести цену ввести «1500».
- 5) При запросе ввести размер туристической группы ввести «10».
- 6) При запросе ввести продолжительность ввести «10 дней».
- 7) При запросе ввести цену ввести «1200».

- 8) При запросе ввести размер туристической группы ввести «12».
- 9) Открыть файл и убедиться, что все записалось.



```
D:\Users\stu-pksp117\source\repos\src\src\bin\Debug\netcoreapp3.1\src.exe
Введите размер массива: 2
Введите продолжительность туристической поездки №1: 14 дней
Введите цену туристической поездки №1: 1500
Введите размер туристической группы №1: 10
Введите продолжительность туристической поездки №2: 10 дней
Введите цену туристической поездки №2: 1200
Введите размер туристической группы №2: 12
Отсортировано!
Сохранение прошло успешно!
```

Рисунок 5 – Результат тестирования

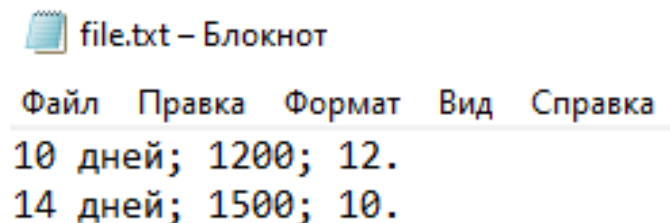


Рисунок 6 – Результат сохранения в файл

Вывод: результаты соответствуют ожидаемым данным.

Тестовый пример 4.

Сценарий:

- 1) Запускаем программу.
- 2) При запросе ввести количества элементов ввести «2».
- 3) При запросе ввести продолжительность ввести «14 дней».
- 4) При запросе ввести цену ввести «1500».
- 5) При запросе ввести размер туристической группы ввести «10».
- 6) При запросе ввести продолжительность ввести «10 дней».
- 7) При запросе ввести цену ввести «1200».
- 8) При запросе ввести размер туристической группы ввести «12».
- 9) Открыть файл и проверить правильности сортировки.


```
D:\Users\stu-pksp117\source\repos\src\src\bin\Debug\netcoreapp3.1\src.exe
Введите размер массива: 2
Введите продолжительность туристической поездки №1: 14 дней
Введите цену туристической поездки №1: 1500
Введите размер туристической группы №1: 10
Введите продолжительность туристической поездки №2: 10 дней
Введите цену туристической поездки №2: 1200
Введите размер туристической группы №2: 12
Отсортировано!
Сохранение прошло успешно!
```

Рисунок 7 – Результат тестирования

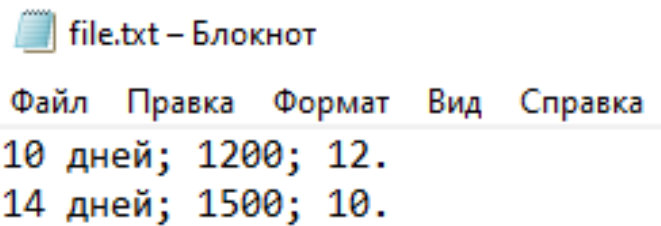


Рисунок 8 – Результат сортировки

Вывод: результаты соответствуют ожидаемым данным.