Тест-требование 1. Проверка некоренного ввода количества туров.

Проверка того, что при вводе пустой строки или ввода отрицательного числа выводит соответствующее сообщение.

Тест-требование 2. Проверка ввода направления поездки.

Проверка того, что при вводе пустой строки выводится соответствующее сообщение.

Тест-требование 3. Проверка некоренного ввода продолжительности поездки.

Проверка того, что при вводе отрицательного числа, 0 или пустой строки выводится соответствующее сообщение.

Тест-требование 4. Проверка некоренного ввода цены поездки.

Проверка того, что при вводе отрицательного числа, 0 или пустой строки выводится соответствующее сообщение.

Тест-требование 5. Проверка корректного ввода количества туров.

Проверка того, что при вводе корректного количества вводимых данных, программа позволяет вводить данные о турах.

Тест-требование 6. Проверка корректного ввода направление поездки.

Проверка того, что после корректного ввода направления поездки программа позволяет вводить данные дальше.

Тест-требование 7. Проверка корректного ввода продолжительности поездки.

Проверка того, что после корректного ввода продолжительности поездки программа позволяет вводить данные дальше.

Тест-требование 8. Проверка корректного ввода цены поездки.

Проверка того, что после корректного ввода цены поездки программа позволяет вводить новые данные о новом туре или сообщается об успешном сохранении данных в файл и производится чтение из файла.

Тест-требование 9. Проверка правильности сортировки по 2 полям.

Проверка того, что при вводе всех данных сортировка правильно сортирует по сочетанию двух свойств по убыванию в приоритете: «Продолжительность», «Цена».

Тест-требование 10. Проверка сохранения отсортированного массива в файл.

Проверка того, что после ввода данных, массив сортируется и записывается в файл.

Тест-план. Тест-требование 1. Проверка некоренного ввода количества туров.

Описание. Проверка того, что при вводе пустой строки или ввода отрицательного числа выводит соответствующее сообщение.

Входные параметры:

* Пустая строка
* Отрицательное число
* 0

Выходные параметры:

* Соответствующее сообщение

Шаги по выполнению:

* запуск программы
* ввод пустой строки
* ввод отрицательного числа
* ввод 0

Тест-план. Тест-требование 2. Проверка ввода направления поездки.

Описание. Проверка того, что при вводе пустой строки выводится соответствующее сообщение.

Входные параметры:

* Пустая строка

Выходные параметры:

* Соответствующее сообщение

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей
* ввод пустой строки

Тест-план. Тест-требование 3. Проверка некоренного ввода продолжительности поездки.

Описание. Проверка того, что при вводе отрицательного числа, 0 или пустой строки выводится соответствующее сообщение.

Входные параметры:

* Пустая строка
* Отрицательное число
* 0

Выходные параметры:

* Соответствующее сообщение

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей
* ввод корректного направление поездки
* ввод 0
* ввод отрицательного числа
* ввод пустой строки

Тест-план. Тест-требование 4. Проверка некоренного ввода цены поездки.

Описание. Проверка того, что при вводе отрицательного числа, 0 или пустой строки выводится соответствующее сообщение.

Входные параметры:

* Пустая строка
* Отрицательное число
* 0

Выходные параметры:

* Соответствующее сообщение

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей
* ввод корректного направление поездки
* ввод корректной продолжительности поездки
* ввод 0
* ввод отрицательного числа
* ввод пустой строки

Тест-план. Тест-требование 5. Проверка корректного ввода количества туров.

Описание. Проверка того, что при вводе корректного количества вводимых данных, программа позволяет вводить данные о турах.

Входные параметры:

* ввод корректных данных (например, 3)

Ожидаемый результат: программа позволяет вводить данные дальше.

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей (например, 3)

Тест-план. Тест-требование 6. Проверка корректного ввода направление поездки.

Описание. Проверка того, что после корректного ввода направления поездки программа позволяет вводить данные дальше.

Входные параметры:

* ввод корректных данных (например, Санкт-Петербург)

Ожидаемый результат: программа позволяет вводить данные дальше.

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей (например, 3)
* ввод Санкт-Петербург

Тест-план. Тест-требование 7. Проверка корректного ввода продолжительности поездки.

Описание. Проверка того, что после корректного ввода продолжительности поездки программа позволяет вводить данные дальше.

Входные параметры:

* ввод корректных данных (например, 1,5)

Ожидаемый результат: программа позволяет вводить данные дальше.

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей (например, 3)
* ввод Санкт-Петербург
* ввод 1,5

Тест-план. Тест-требование 8. Проверка корректного ввода цены поездки.

Описание. Проверка того, что после корректного ввода цены поездки программа позволяет вводить новые данные о новом туре или сообщается об успешном сохранении данных в файл и производится чтение из файла.

Входные параметры:

* ввод корректных данных (например, 100000)

Ожидаемый результат: программа либо позволяет вводить данные дальше, либо происходит запись в файл и чтение из файла, данные из файла выводятся в консоли.

Шаги по выполнению:

* запуск программы
* ввод корректного количества записей (например, 3)
* ввод Санкт-Петербург
* ввод 1,5
* ввод 100000

Тест-план. Тест-требование 9. Проверка правильности сортировки по 2 полям.

Описание. Проверка того, что при вводе всех данных сортировка правильно сортирует по сочетанию двух свойств по убыванию в приоритете: «Продолжительность», «Цена».

Входные параметры 1:

* Количество записей 2
* Направление поездки: Санкт-Петербург, Москва
* Продолжительность поездки: 3, 4.5
* Цена поездки: 100000, 150000

Ожидаемый результат:

* 1 – я запись Москва, 4.5, 150000
* 2 – я запись Санкт-Петербург, 3, 100000

Входные параметры 2:

* Количество записей 2.
* Направление поездки: Владимир, Санкт-Петербург.
* Продолжительность поездки: 3, 3.
* Цена поездки: 159000, 160000.

Ожидаемый результат:

* 1 – я запись Санкт-Петербург, 3, 160000
* 2 – я запись Владимир, 3, 159000

Тест-план. Тест-требование 10. Проверка сохранения отсортированного массива в файл.

Описание. Проверка того, что после ввода данных, массив сортируется и записывается в файл.

Входные параметры:

* Корректные данные о турах

Ожидаемый результат: после ввода данных все должно записаться в файл «TourList», который находит в корневой папке проекта: «Exam\_martyuk\Exam\_martyuk\bin\Debug\netcoreapp3.1».

Результаты тестирования:

|  |  |  |
| --- | --- | --- |
| Тест 1 | Проверка на некорректный ввод количества записей | Соответствующее сообщение о неверном вводе количества записей выводится |
| Тест 2 | Проверка на некорректный ввод направление поездки | Соответствующее сообщение о пустой строке выводится |
| Тест 3 | Проверка на некорректный ввод продолжительности поездки | Соответствующее сообщение о неверном вводе продолжительности поездки выводится |
| Тест 4 | Проверка на некорректный ввод цены поездки | Соответствующее сообщение о неверном вводе цены поездки выводится |
| Тест 5 | Проверка на корректный ввод количества записей | Программа позволяет вводить данные дальше |
| Тест 6 | Проверка на корректный ввод направление поездки | Программа позволяет вводить данные дальше |
| Тест 7 | Проверка на корректный ввод продолжительности поездки | Программа позволяет вводить данные дальше |
| Тест 8 | Проверка на некорректный ввод цены поездки | Программа позволяет вводить данные дальше или завершается сообщением о успешной записи данных в файл и вывод данных из файла |
| Тест 9 | Проверка правильности сортировки по условию задачи | После ввода данных сортировка производится по 2 полям успешно |
| Тест 10 | Проверка записи данных в файл. | После ввода данных при открытии файла по пути «Exam\_martyuk\Exam\_martyuk\bin\Debug\netcoreapp3.1» там действительно отсортированные есть данные. |