|  |  |
| --- | --- |
| Программа для определения, являются ли все числа в последовательности числами Фибоначчи | |
| Программа и методика испытаний | |
| Студент | Вольвич Д. С. |
| Преподаватель | Якимов Г. А. |
| Сдано |  |

1 Объект испытаний

Наименование программы: “FibonacciNumbers” (“Программа для определения, является ли заданная последовательность чисел числами Фибоначчи”).

Программа написана на языке C++.

2 Цель испытаний

Целью испытаний является проверка соответствия программы

требованиям к функциональным характеристикам.

3 Требования к программе

Требования к программе изложены в Техническом задании в пункте 3.

4 Требования к программной документации

В бумажной форме должны быть предоставлены: техническое задание, технический проект в виде описания программы, программа и методика испытаний и руководство программиста. В электронном формате должны быть предоставлены: копии всех документов бумажной формы, рабочая документация и текст программы. Вся документация должна быть составлена в соответствии с ГОСТ 19.

5 Средства и порядок испытаний

Для запуска тестов необходимо подключение библиотеки CppUnitTest к проекту программы.

6 Методы испытаний

Аспекты тестирования приведены в приложении 1. Перечень проверок для функции isValidLine приведен в приложении 2, для функции validateInput в приложении 3, для функции isFibonacciSequence в приложении 4.

Приложение 1

Аспекты тестирования (Таблица №1)

|  |  |  |
| --- | --- | --- |
| Тестируемая функция | Аспекты тестирования | Варианты значений |
| isValidLine | Наличие некорректных символов в строке | Строка содержит только цифры и пробелы;  Строка содержит только некорректные символы;  Строка содержит и корректные символы (цифры и пробелы), и некорректные. |
| validateInput | Различные варианты строки во входном файле | Ровно одна корректная строка с последовательностью Фибоначчи;  Ровно одна корректная строка, не содержащая последовательность Фибоначчи;  Пустая строка;  Две строки, содержащие последовательность Фибоначчи;  Две строки, не содержащие последовательность Фибоначчи;  Ровно одна некорректная строка;  Строка содержит отрицательные числа;  Строка содержит только одно число;  Строка содержит 1001 число. |
| isFibonacciSequence | Различные варианты корректных строк | Минимальная последовательность Фибоначчи;  Последовательность Фибоначчи из 5 чисел;  Последовательность Фибоначчи из 20 чисел;  Минимальная строка, не содержащая последовательность Фибоначчи;  Строка из 5 чисел с 3-им элементом, нарушающим последовательность Фибоначчи;  Строка из 20 чисел с 10-ым элементом, нарушающим последовательность Фибоначчи;  Строка из 7 чисел, с двумя идущими подряд (3-ий и 4-ый) элементами, нарушающими последовательность Фибоначчи. |

Приложение 2

Проверка, что строка содержит только цифры и пробелы.

bool isValidLine(const string& line)

Входные данные:

* line – строка из входного файлы.

Выходные данные:

* false – если содержит некорректные символы; true – если содержит только цифры и пробелы.

Таблица №2. Тесты для функции isValidLine

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название тестов | Тестовые данные | Ожидаемый результат |
| 1 | Строка содержит только цифры и пробелы | 0 1 1 2 3 | true |
| 2 | Строка содержит только некорректные символы | \* \_ ! | false |
| 3 | Строка содержит и корректные символы (цифры и пробелы), и некорректные | 0 1 1 \* 3 | false |

Приложение 3

Проверка корректности входного файла.

validateInput(ifstream& inputFile, ofstream& outputFile, vector<int>& sequence)

Входные данные:

* inputFile – входной файл;
* outputFile – выходной файл;
* sequence – последовательность чисел.

Выходные данные:

* 0 – нет ошибок; 3 – входной файл пуст; 4 – более одной строки; 5 – недопустимые символы; 6 – меньше 2 чисел; 7 – больше 1000 чисел.

Таблица №3. Тесты для функции validateInput

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название тестов | Тестовые данные | Ожидаемый результат |
| 1 | Ровно одна корректная строка с последовательностью Фибоначчи | 0 1 1 2 3 5 | 0 |
| 2 | Ровно одна корректная строка, не содержащая последовательность Фибоначчи | 0 1 1 3 | 0 |
| 3 | Пустая строка |  | 3 |
| 4 | Две строки, содержащие последовательность Фибоначчи | 0 1 1 2  0 1 1 2 3 | 4 |
| 5 | Две строки, не содержащие последовательность Фибоначчи | 0 1 1 3  0 1 1 2 5 | 4 |
| 6 | Ровно одна некорректная строка | 0 1 1 \* 3 | 5 |
| 7 | Строка содержит отрицательные числа | -2 -1 0 1 2 | 5 |
| 8 | Строка содержит только одно число | 1 | 6 |
| 9 | Строка содержит 1001 число | 1 1 1 1 1 1 1 … 1 1 1 | 7 |

Приложение 4

Проверка на последовательность чисел Фибоначчи.

Bool isFibonacciSequence(const vector<int>& sequence, int& errorIndex)

Входные данные:

* sequence – последовательность чисел;
* errorIndex – индекс первого ошибочного элемента.

Выходные данные:

* false – если последовательность чисел не соответствует последовательности Фибоначчи; true – если последовательность чисел соответствует последовательности Фибоначчи.

Таблица №4. Тесты для функции isFibonacciSequence

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название тестов | Тестовые данные | Ожидаемый результат |
| 1 | Минимальная последовательность Фибоначчи | 0 1 | true |
| 2 | Последовательность Фибоначчи из 5 чисел | 0 1 1 2 3 | true |
| 3 | Последовательность Фибоначчи из 20 чисел | 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610  987 1597 2584 4181 | true |
| 4 | Минимальная строка, не содержащая последовательность Фибоначчи | 1 1 | false |
| 5 | Строка из 5 чисел с 3-им элементом, нарушающим последовательность Фибоначчи | 0 1 2 2 3 | false |
| 6 | Строка из 20 чисел с 10-ым элементом, нарушающим последовательность Фибоначчи | 0 1 1 2 3 5 8 13 21 45 55 89 144 233 377 610  987 1597 2584 4181 | false |
| 7 | Строка из 7 чисел, с двумя идущими подряд (3-ий и 4-ый) элементами, нарушающими последовательность Фибоначчи | 0 1 2 3 3 5 8 | false |