|  |  |
| --- | --- |
| Программа для вывода последовательности простых чисел | |
| Программа и методика испытаний | |
| Студент | Вихлянцев В.В. |
| Преподаватель | Кулюкин К.С. |
| Сдано |  |

1. Объект испытаний

Наименование программы: "PrimeNumbersFinder" ("Программа для нахождения простых чисел до заданного числа N").

Программа разработана на языке C++, соответствует требованиям, изложенным в техническом задании, внешней и внутренней спецификации, и предназначена для работы в операционной системе Windows 10 или выше.

1. Цель испытаний

Целью испытаний является проверка соответствия программы требованиям к функциональным характеристикам, включая корректность работы алгоритма поиска простых чисел и обработки входных данных.

1. Требования к программе

Требования к программе изложены в Техническом задании в пункте 3.

1. Требования к программной документации

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

1. Средства и порядок испытаний

Для запуска тестов необходимо подключение библиотеки CppUnitTest к проекту программы.

1. Метод испытаний

Перечень проверок для функции isInputValid приведен в приложении 1, для функции findPrimes – в приложении 2.

Приложение 1

Методика тестирования функции isInputValid

/\*!

\* \Проверяет корректность входных данных и преобразует строку в число

\* \param[in] line - строка из входного файла

\* \param[out] N - преобразованное число (только при успешной валидации)

\* \param[out] error - структура с информацией об ошибке (при неудачной валидации)

\* \return true - если данные корректны, false - если есть ошибки

\* \throw - не выбрасывает исключения

\*/bool isInputValid(const string& line, int& N, ErrorInfo& error);

Входные данные:

line – строка, содержащая потенциальное число N.

N – выходная переменная для сохранения числа (только при успешной валидации).

errorMessage – выходная структура для информации об ошибке (заполняется при ошибке).

Выходные данные:

Возвращает true, если:

1. Строка не пустая после тримминга.
2. Содержит только цифры.
3. Число находиться в диапазоне [2, 10000).

Возвращает false, если данные некорректны.

Заполняет структуру error в случае ошибки:

1. EMPTY\_INPUT(3): Входной файл пуст. Введите натуральное число N.
2. INVALID\_FORMAT(5): Ожидается натуральное число в диапазоне [2, 10000).
3. OUT\_OF\_RANGE(6): Число N должно быть в диапазоне [2, 10000).

Таблица №1. Тесты для функции isInputValid.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название теста | Входная строка (line) | Ожидаемый (N) | Ожидаемый (error.code) |
| 1 | Корректное число | "42" | 42 | - |
| 2 | Числа с пробелами | " 100 " | 100 | - |
| 3 | Пустая строка | "" | - | EMPTY\_INPUT(3) |
| 4 | Строка из пробелов | " " | - | EMPTY\_INPUT(3) |
| 5 | Текст вместо числа | "abc" | - | INVALID\_FORMAT(5) |
| 6 | Число с буквами | "12a34" | - | INVALID\_FORMAT(5) |
| 7 | Нижняя граница (недопустима) | "1" | - | 6 |
| 8 | Минимальное допустимое значение | "2" | 2 | - |
| 9 | Максимальное допустимое значение | "9999" | 9999 | - |
| 10 | Верхняя граница (недопустимая) | "10000" | - | OUT\_OF\_RANGE(6) |
| 11 | Ноль | "0" | - | OUT\_OF\_RANGE(6) |
| 12 | Отрицательное число | "-5" | - | INVALID\_FORMAT(5) |
| 13 | Дробное число | "2.5" | - | INVALID\_FORMAT(5) |
| 14 | Число с ведущими нулями | "007" | 7 | - |
| 15 | Число с разделителями | "1,03" | - | INVALID\_FORMAT(5) |

Приложение 2

Методика тестирования функции findPrimes

/\*!

\* \Находит все простые числа до заданного N (Решето Эратосфена)

\* \param[in] N - верхняя граница поиска

\* \return set<int> - множество найденных простых чисел в порядке возрастания

\* \note Для N < 2 возвращает пустое множество

\* \throw - не выбрасывает исключения

\*/

set<int> findPrimes(int N);

Входные данные:

N – целое число, верхняя граница поиска (2 <= N < 10000).

Выходные данные:

Возвращает множество (set<int>) простых чисел от 2 до N включительно.

Для N < 2 возвращает пустое множество.

Гарантируется сортировка чисел (т.к. используется set)

Таблица №2. Тесты для функции findPrimes.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название теста | Входное число (N) | Ожидаемый результат (простые числа) |
| 1 | Минимальное допустимое N | 2 | {2} |
| 2 | Пустая строка | "" | {} |
| 3 | Строка из пробелов | " " | {} |
| 4 | Текст вместо числа | "abc" | {} |
| 5 | Число с буквами | "12a34" | {} |
| 6 | Отрицательное число | "-5" | {} |
| 7 | Дробное число | "2.5" | {} |
| 8 | Спецсимволы | "#@!100" | {} |
| 9 | Число меньше допустимого | "1" | {} |
| 10 | Число больше допустимого | "10000" | {} |
| 11 | Число с ведущими нулями | "007" | {2, 3, 5, 7} |
| 12 | Ноль | "0" | "0" |
| 13 | Максимально допустимое значение | "9999" | {2, 3, 5, ... , 9973} |
| 14 | Число с разделителями | "1,000" | {} |
| 15 | Обычное число | "11" | {2, 3, 5, 7, 11} |