Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Преподаватель |  |  |  |  | Р.Ю. Царев |
|  |  |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ19–04–1М, 031943354 |  |  |  | Т.В. Радионов |
|  | номер группы, зачетной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2020

# Цель и задачи

*Цель:*

Применить метод ввода ошибок, анализируя надежность программы N-версии из лаборатории № 1.

*Задачи:*

* описать реализованный метод внедрения ошибок программного обеспечения;
* уметь объяснить, как оценивается вероятность отказа по требованию;
* предоставить анализ полученного результата

# Описание программы

В данной работе описывается проблема поиска минимума функции различными методами. В качестве исходных данных взята функция 1 и три алгоритма поиска экстремумов для этой функции: метод покоординатного спуска, метод Флетчера-Ривса и метод Дэвидона-Флетчера-Пауэлла.

|  |  |
| --- | --- |
|  | (1) |

Для каждого метода поиска экстремума было произведено по 5 вычислений при различных параметрах и вычислена медиана среди этих вычислений (для точности результата и исключения аномальных результатов).

В качестве алгоритма голосования для N-версии было выбрано усреднение.

# Описание метода внедрения ошибок

Ошибка добавляется путем задания свойства IsError = true для какой-либо версии после формирования списка результатов метода этой версии.

Далее производится вызов метода AddError с передачей списка для ввода ошибок путем случайной генерацией чисел.

Формула ошибки для каждого выхода представлена в формуле 2, где с помощью генератора случайных чисел определяется во сколько раз увеличить/уменьшить выход результата (D – максимальное число для генерации или во сколько раз увеличится значение выхода).

|  |  |
| --- | --- |
|  | (2) |

Далее путем медианного исключения выбросов и алгоритмом голосования рассчитывается итоговый результат вычисления функции.

Если полученный результат отличается от истинного результата (без внедрения ошибки) более чем на 1 (погрешность для координаты X), то считается, что данная программа дала неверный результат, что приравнивается к сбою (Error).

Для анализа надежности (POFOD) воспользуемся формулой 3, где Q – вероятность отказа, P – вероятность безотказной работы, N – количество запусков программы, Error – количество сбоев программы.

|  |  |
| --- | --- |
|  | (3) |

# Ход работы

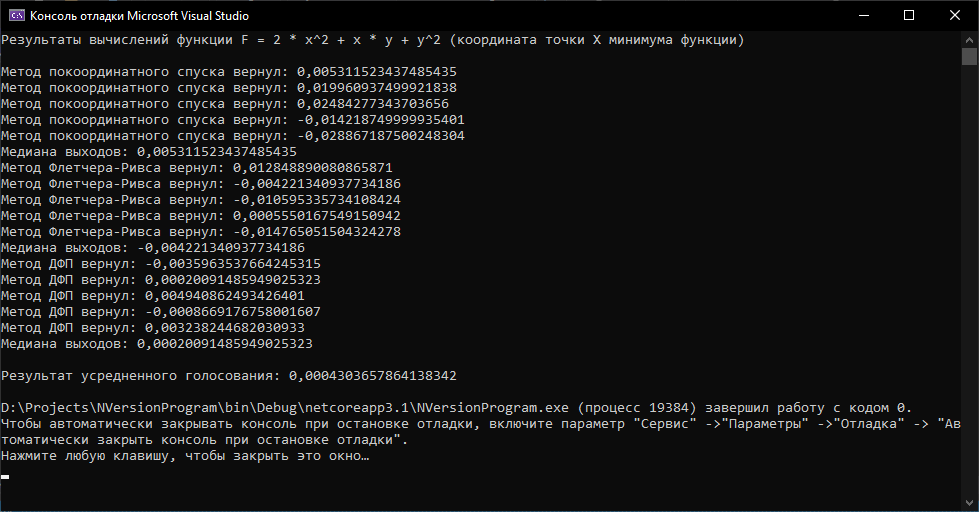


Рисунок 1 – Вывод вычислений и результатов программы беш внедрения ошибки

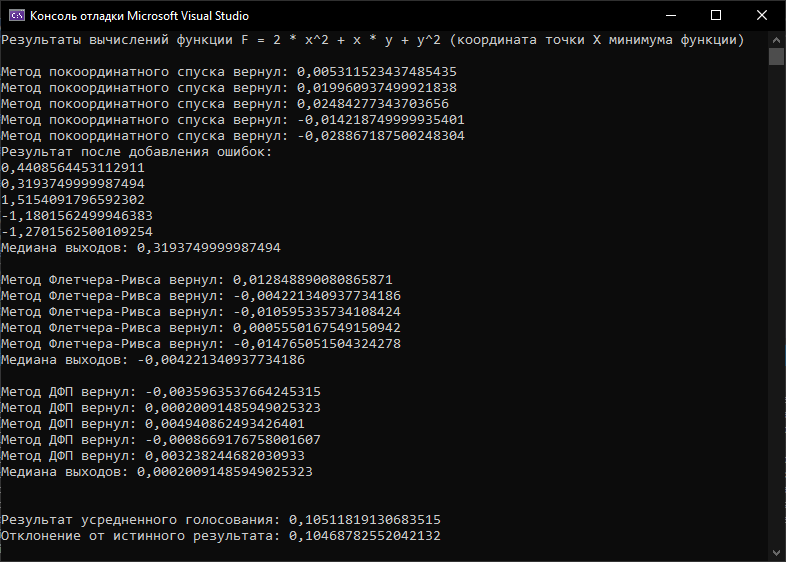


Рисунок 2 – Внедрение ошибки в первый алгоритм (D = 100)

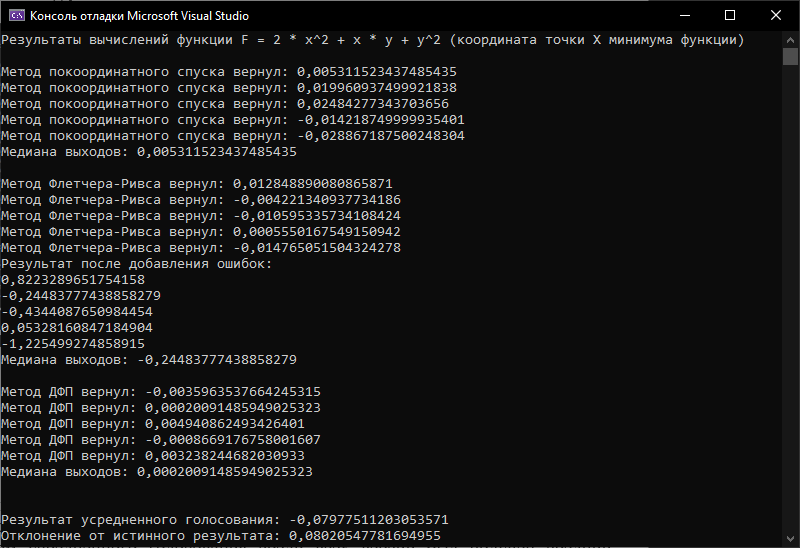


Рисунок 3 – Внедрение ошибки во второй алгоритм (D = 100)

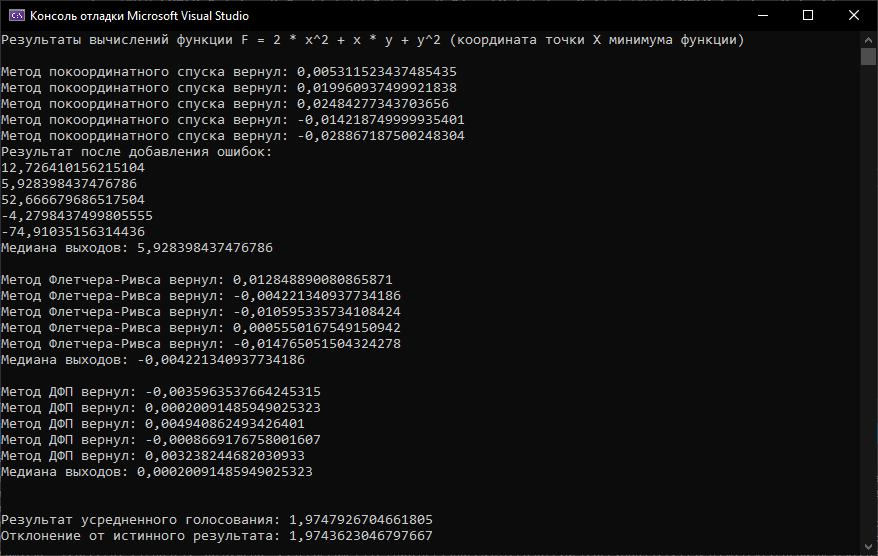


Рисунок 4 – внедрение ошибки в первый алгоритм (D = 3000)

# Расчет надежности программы

Вычислим надежность программы по формуле 3 при D = 100, 1000, 5000, 10000 (см. формула 2) и проведем 10 запусков (N) для каждого случая для определения вероятности отказа (Q). Внедрение ошибки будем осуществлять в первый алгоритм.

Таблица 1 – Результаты надежности программы (Q)

|  |  |  |  |
| --- | --- | --- | --- |
| D = 100 | D = 1000 | D = 5000 | D = 10000 |
| 0 | 0,3 | 0,8 | 0,9 |

# Выводы

Усредненное голосование не самый надежный метод для повышения надежности программы в плане того, что если какая-либо версия даст абсолютный сбой, то это приведет к отказу всей системы, т.к. среднее значение при голосовании, например, между 0.1, 0.2, 1000 явно будет гораздо более отдаленное от 0 из-за 1000 в последней версии. Однако усредненное голосование позволило снизить вероятность отказа системы до 0 при отклонении результата в какой-либо версии в 100 раз. Сбои в программе начались лишь при отклонении результатов в 1000 и более раз.