|  |
| --- |
| Министерство науки и высшего образования РФ Федеральное государственное автономное образовательное учреждение высшего профессионального образования **«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»** |
| Институт космических и информационных технологий |
| Кафедра вычислительной техники |

**Практическая работа №5**

**«**Инструментальные средства анализа кода программ**»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Преподаватель |  |  |  |  |  | Л.С. Артемьев |
|  |  |  |  |  | подпись, дата |  |
| Студент |  | КИ22-07Б, 032212158 |  |  | А.С. Чернов |
|  |  | номер групп, зачетной книжки |  | подпись, дата |  |

Красноярск 2023

**Техническое задание**

1. Исследование программы полученной в ходе выполнения работы 3 с помощью инструментов статического и динамического анализа кода.
2. Оценка полученных решений в соответствии с критериями качества ПО.
3. Улучшение программы – устранение обнаруженных недочётов.
4. Исследование программы с использованием статического анализатора

Исследование будет проводиться с использование статического анализатора PVS-Studio. После анализа нашего кода получаем результат, показанный на рисунке 1.

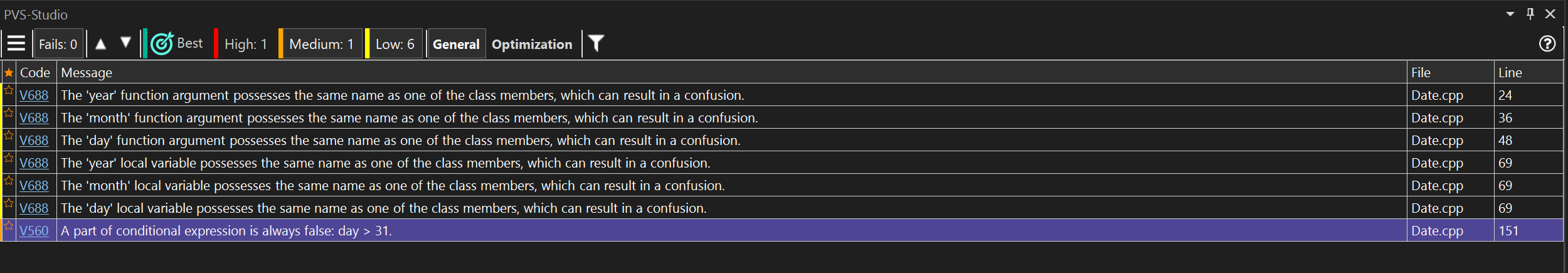


Рисунок 1 – Результат проверки в PVS-Studio

**Устранение полученных недочетов**

Для решения “Medium” ошибки, обнаруженной статическим анализатором PVS-Studio, я изменил условие в методе Date::IsValidDate(). В частности, я уточнил условия проверки дня в месяце, чтобы избежать ситуации, когда day больше 31, что может привести к тому, что условие всегда будет ложным.

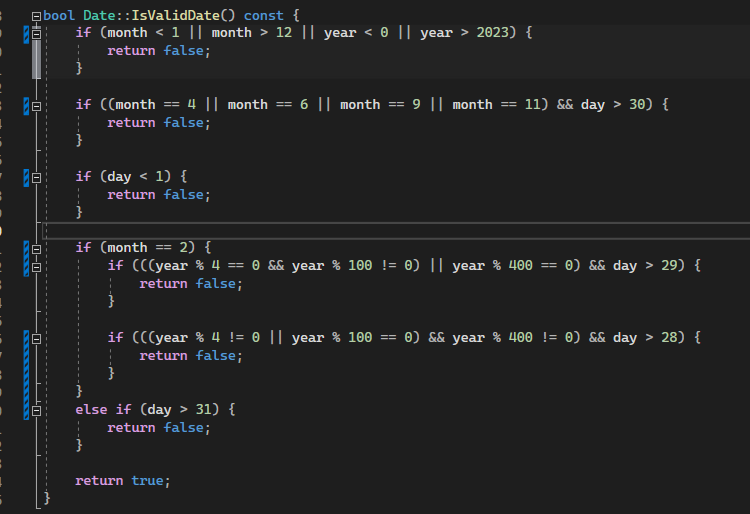


Рисунок 2 – Исправленный вариант кода.

Так как устранение “Low” ошибок не требует больших усилий я покажу только устранение нескольких из них.

Ошибка V688 указывает на то, что имя аргумента функции совпадает с именем члена класса, что может привести к путанице. В данном случае, это относится к методу **Date::SetYear**. Для устранения этой ошибки, надо изменить имя аргумента, чтобы оно отличалось от имени члена класса.

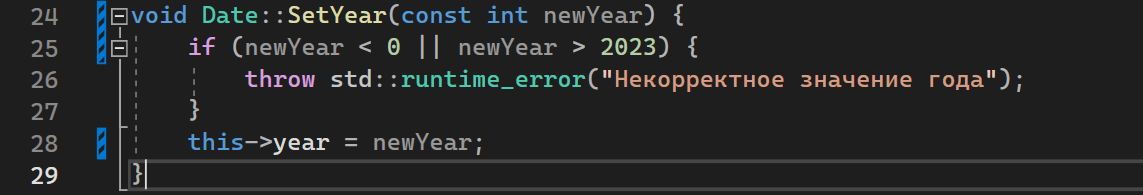


Рисунок 3 – Исправленный вариант кода для одной из “Low” ошибок.

В этом случае, ошибка V688 указывает на конфликт имен между локальными переменными year, month, day и членом класса year, month, day. Для разрешения этой ошибки нужно изменить имя локальной переменной, чтобы оно отличалось от имени члена класса.

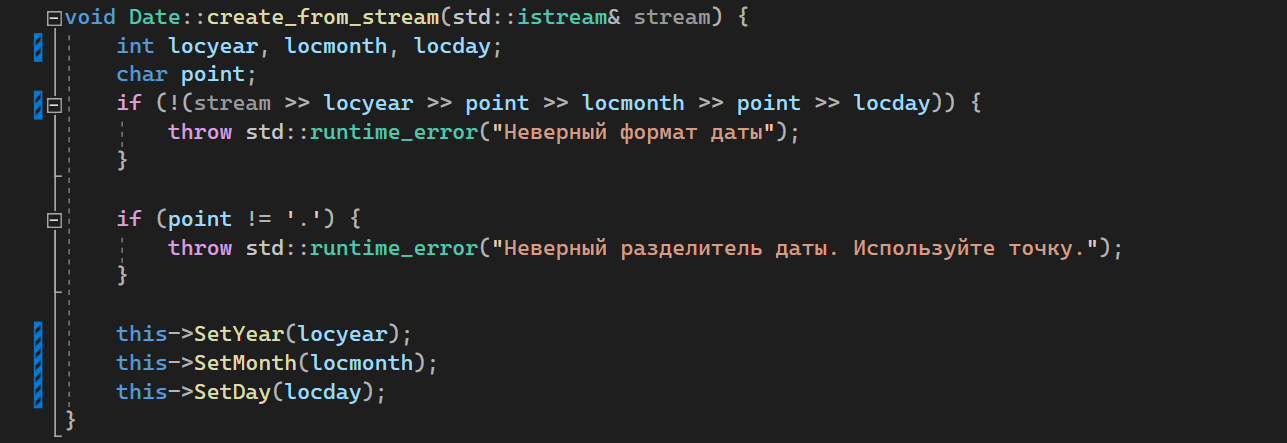


Рисунок 3 – Исправленный вариант кода для оставшихся “Low” ошибок.

Метод “SetDay” проверяет, что значение дня находится в допустимом диапазоне от 1 до 31. Однако, не учитывается количество дней в конкретном месяце.

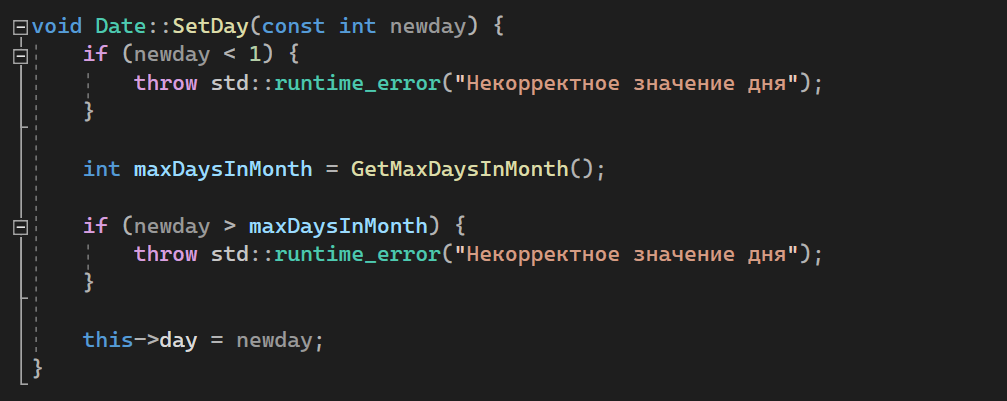


Рисунок 4 – Исправленный вариант кода для метода Set\_day с использованием функции “GetMaxDaysInMonth”.

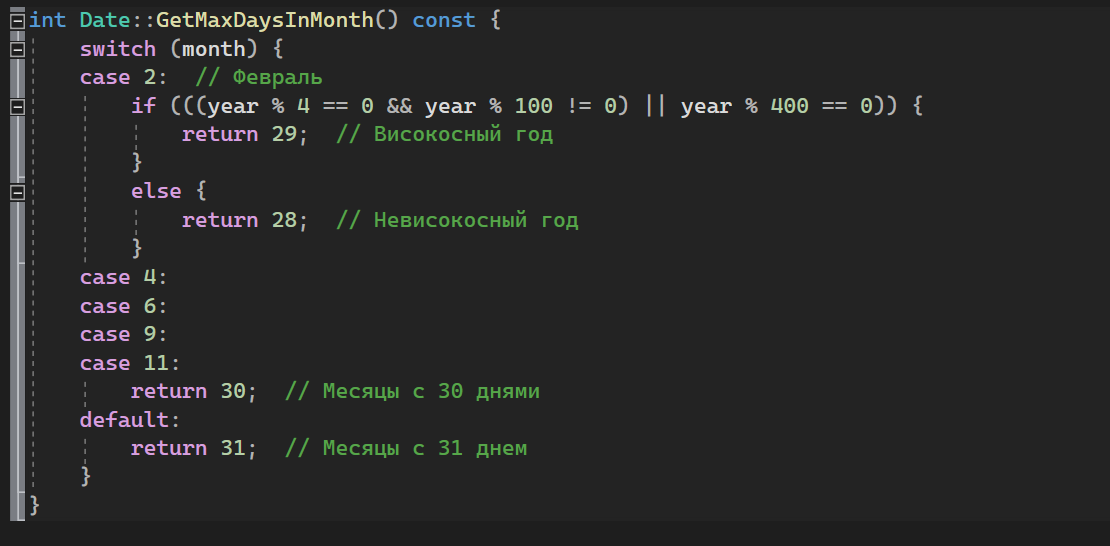


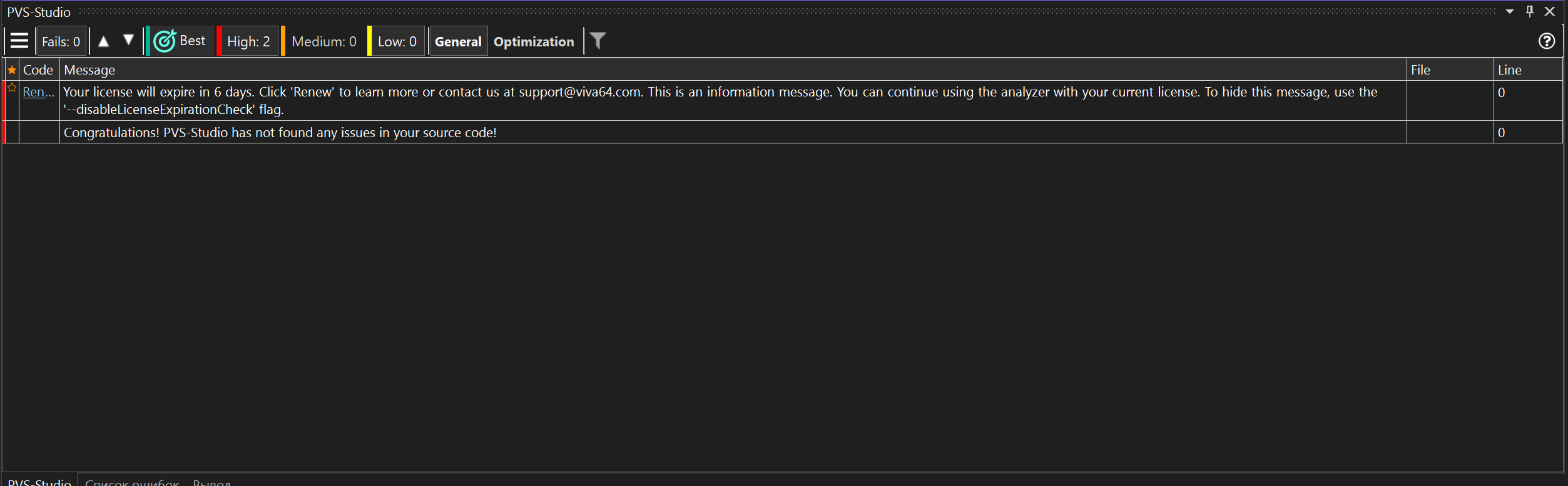
Рисунок 4 –Функция “GetMaxDaysInMonth”.

Валидаторы в файле “Student\_topic.cpp” (ValidateName и ValidateTopicName) определены, но не используются в методах “Set\_\_\_” и конструкторе



Рисунок 5 – Исправленный вариант кода.

После устранения всех ошибок запускаем анализатор кода опять и видим:



Оставшиеся ошибки указывают на то, что у меня осталось 6 дней пробной версии анализатора кода. Соответственно мы исправили все ошибки по коду.

1. Исследование программы с использование динамического анализатора

Воспользуемся динамическим анализатором CRT для проверки на утечки памяти. Для этого добавим макросы в самое начало в основном файле, а в конце работы программы вызовем функцию \_CrtDumpMemoryLeaks().

Изменения можно посмотреть в фиксации с номером fb06e8d4.

После выполнения программы не получаем сообщений о найденных утечках, а значит программа не имеет утечек памяти.

**Заключение**

В ходе выполнения практической работы мы воспользовались статическим анализатором PVS-Studio который помог нам обнаружить ошибки, которые мы исправили в процессе выполнения работы. Также мы воспользовались динамическим анализатором, который помогает отследить утечки памяти, в нашем случае утечек памяти выявлено не было. Поставленное изначально задание было выполнено.