Отчёт по использованию динамических библиотек

# Преимущества и недостатки динамического связывания

Преимущества:

* - Меньший размер исполняемого файла, так как код библиотеки не включается напрямую.
* - Возможность обновления библиотеки без перекомпиляции основного приложения.
* - Повторное использование кода в нескольких приложениях.

Недостатки:

* - Необходимость наличия библиотеки в системе при запуске.
* - Возможные проблемы с совместимостью версий библиотек.
* - Сложности в отладке и настройке окружения.

# Когда предпочтительно использовать динамические библиотеки

- При разработке больших проектов, где несколько приложений используют общий функционал.  
- В случае, если библиотека часто обновляется и необходимо минимизировать распространение новых версий.  
- При создании модульных систем, где компоненты могут подключаться или обновляться динамически.

# Влияние на размер исполняемого файла и использование памяти

Использование динамических библиотек позволяет значительно уменьшить размер исполняемого файла, так как библиотека не включается напрямую в бинарник. Вместо этого она подключается во время выполнения.  
Также это может снизить общее потребление памяти, если несколько процессов используют одну и ту же библиотеку, поскольку её код может быть загружен в память один раз и разделён между процессами.

# Анализ и улучшения кода

Возможные улучшения:

* - Добавить проверку на переполнение при арифметических операциях.
* - Вынести сообщения об ошибках в отдельный модуль или макросы.

Дополнительные функции для библиотеки:

* - Деление двух чисел с проверкой деления на ноль.
* - Возведение в степень.
* - Функции для работы с массивами (сумма элементов, среднее значение и т.п.).