

Качество программного обеспечения — способность программного продукта при заданных условиях удовлетворять установленным или предполагаемым потребностям^[1], весь объем признаков и характеристик программ, который относится к их способности удовлетворять установленным или предполагаемым потребностям^{[2][3]}.

Качество кода может определяться различными критериями. Некоторые из них имеют значение только с точки зрения человека. Например, то, как отформатирован текст программы, совершенно не важно для компьютера, но может иметь серьёзное значение для последующего сопровождения. Многие из имеющихся [стандартов оформления кода](#), определяющих специфичные для используемого языка соглашения и задающие ряд правил, улучшающих читаемость кода, имеют своей целью облегчить будущее сопровождение ПО, включающее отладку и обновление. Существуют и другие критерии, определяющие, «хорошо» ли написан код, например, такие, как структурированность — степень логического разбиения кода на ряд управляемых блоков.

- Читаемость кода
- Лёгкость поддержки, [тестирования](#), отладки, исправления ошибок, изменения и [портируемости](#)
- Низкая сложность кода
- Низкое использование ресурсов: памяти и процессорного времени
- Корректная обработка [исключительных ситуаций](#)
- Малое число предупреждений при [компиляции](#) и [линковке](#)

Фактор качества ПО — это нефункциональное требование к программе, которое обычно не описывается в договоре с заказчиком, но, тем не менее, является желательным требованием, повышающим качество программы.

Некоторые из факторов качества:

понятность

Назначение ПО должно быть понятным, из самой программы и документации.

полнота

Все необходимые части программы должны быть представлены и полностью реализованы.

краткость

Отсутствие лишней, дублирующейся информации. Повторяющиеся части кода должны быть преобразованы в вызов общей процедуры. То же касается и документации.

портируемость

Лёгкость в адаптации программы к другому окружению: другой архитектуре, платформе, операционной системе или её версии.

согласованность

По всей программе и в документации должны использоваться одни и те же соглашения, форматы и обозначения.

сопровождаемость

Насколько сложно изменить программу для удовлетворения новых требований. Это требование также указывает, что программа должна быть хорошо документирована, не слишком запутана, и иметь резерв роста по использованию ресурсов (память, процессор).

тестируемость

Позволяет ли программа выполнить проверку приёмочных характеристик, поддерживается ли возможность измерения производительности.

удобство использования
Простота и удобство использования программы. Это требование относится прежде всего к интерфейсу пользователя.

надёжность
отсутствие отказов и сбоев в работе программ, а также простота исправления дефектов и ошибок.

структурированность

эффективность
Насколько рационально программа относится к ресурсам (память, процессор) при выполнении своих задач.

безопасность

С точки зрения пользователя

Помимо технического взгляда на качество ПО, существует и оценка качества с позиции пользователя. Для этого аспекта качества иногда используют термин «[юзабилити](#)». Довольно сложно получить оценку юзабилити для заданного программного продукта. Наиболее важные из вопросов, влияющий на оценку:

- Является ли [пользовательский интерфейс](#) интуитивно понятным?
- Насколько просто выполнять простые, частые операции?
- Насколько легко выполняются сложные операции?
- Выдаёт ли программа понятные [сообщения об ошибках](#)?
- Всегда ли программа ведёт себя так как ожидается?
- Имеется ли [документация](#) и насколько она полна?
- Является ли интерфейс пользователя само-описательным/само-документирующим?
- Всегда ли задержки с ответом программы являются приемлемыми?