

Методы поиска ошибок и процедура тестирования

Международный стандарт ANSI/IEEE-729-83 разделяет все ошибки в разработке программ на следующие типы.

Ошибка (error) - состояние программы, при котором выдаются неправильные результаты, причиной которых являются изъяны (*flaw*) в операторах программы или в технологическом процессе ее разработки, что приводит к неправильной интерпретации исходной информации, следовательно, и к неверному решению.

Дефект (fault) в программе - следствие ошибок разработчика на любом из этапов разработки, которая может содержаться в исходных или проектных спецификациях, текстах кодов программ, эксплуатационной документации и т.п. В процессе выполнения программы может быть обнаружен дефект или сбой.

Отказ (failure) - это отклонение программы от функционирования или невозможность программы выполнять функции, определенные требованиями и ограничениями, что рассматривается как событие, способствующее переходу программы в неработоспособное состояние из-за ошибок, скрытых в ней дефектов или сбоев в среде функционирования [7.6, 7.11]. Отказ может быть результатом следующих причин:

- ошибочная спецификация или пропущенное требование, означающее, что спецификация точно не отражает того, что предполагал пользователь;
- спецификация может содержать требование, которое невозможно выполнить на данной аппаратуре и программном обеспечении;
- проект программы может содержать ошибки (например, база данных спроектирована без средств защиты отнесанкционированного доступа пользователя, а требуется защита);

- программа может быть неправильной, т.е. она выполняет несвойственный алгоритм или он реализован не полностью.

Таким образом, отказы, как правило, являются результатами одной или более ошибок в программе, а также наличия разного рода дефектов.

Ошибки на этапах процесса тестирования. Приведенные типы ошибок распределяются по этапам ЖЦ и им соответствуют такие источники их возникновения [7.12]:

- непреднамеренное отклонение разработчиков от рабочих стандартов или планов реализации;
- спецификации функциональных и интерфейсных требований выполнены без соблюдения стандартов разработки, что приводит к нарушению функционирования программ;
- организации процесса разработки - несовершенная или недостаточное управление руководителем проекта ресурсами (человеческими, техническими, программными и т.д.) и вопросами тестирования и интеграции элементов проекта.

Рассмотрим процесс тестирования, исходя из рекомендаций стандарта ISO/IEC 12207, и приведем типы ошибок, которые обнаруживаются на каждом процессе ЖЦ.

Процесс разработки требований. При определении исходной концепции системы и исходных требований к системе возникают ошибки аналитиков при спецификации верхнего уровня системы и построении концептуальной модели предметной области.

Характерными ошибками этого процесса являются:

- неадекватность спецификации требований конечным пользователям;- некорректность спецификации взаимодействия ПО со средой функционирования или с пользователями;

- несоответствие требований заказчика к отдельным и общим свойствам ПО;
- некорректность описания функциональных характеристик;
- необеспеченность инструментальными средствами всех аспектов реализации требований заказчика и др.

Процесс проектирования. Ошибки при проектировании компонентов могут возникать при описании алгоритмов, логики управления, структур данных, интерфейсов, логики моделирования потоков данных, форматов ввода-вывода и др. В основе этих ошибок лежат дефекты спецификаций аналитиков и недоработки проектировщиков. К ним относятся ошибки, связанные:

- с определением интерфейса пользователя со средой;
- с описанием функций (неадекватность целей и задач компонентов, которые обнаруживаются при проверке комплекса компонентов);
- с определением процесса обработки информации и взаимодействия между процессами (результат некорректного определения взаимосвязей компонентов и процессов);
- с некорректным заданием данных и их структур при описании отдельных компонентов и ПС в целом;
- с некорректным описанием алгоритмов модулей;
- с определением условий возникновения возможных ошибок в программе;
- с нарушением принятых для проекта стандартов и технологий.

Этап кодирования. На данном этапе возникают ошибки, которые являются результатом дефектов проектирования, ошибок программистов и

менеджеров в процессе разработки и отладки системы. Причиной ошибок являются:

- бесконтрольность значений входных параметров, индексов массивов, параметров циклов, выходных результатов, деления на 0 и др.;
- неправильная обработка нерегулярных ситуаций при анализе кодов возврата от вызываемых подпрограмм, функций и др.;
- нарушение стандартов кодирования (плохие комментарии, нерациональное *выделение модулей* и компонент и др.);
- использование одного имени для обозначения разных объектов или разных имен одного объекта, плохая мнемоника имен;- несогласованное внесение изменений в программу разными разработчиками и др.

Процесс тестирования. На этом процессе ошибки допускаются программистами и тестировщиками при выполнении технологии сборки и тестирования, выбора тестовых наборов и сценариев тестирования и др. Отказы в программном обеспечении, вызванные такого рода ошибками, должны выявляться, устраняться и не отражаться на статистике ошибок компонент и программного обеспечения в целом.

Процесс сопровождения. На процессе сопровождения обнаруживаются ошибки, причиной которых являются недоработки и дефекты эксплуатационной документации, недостаточные показатели модифицируемости и удобочитаемости, а также некомпетентность лиц, ответственных за сопровождение и/или усовершенствование ПО. В зависимости от сущности вносимых изменений на этом этапе могут возникать практически любые ошибки, аналогичные ранее перечисленным ошибкам на предыдущих этапах.

Все ошибки, которые возникают в программах, принято подразделять на следующие классы [7.12]:

- логические и функциональные ошибки;
- ошибки вычислений и времени выполнения;
- ошибки ввода-вывода и манипулирования данными;
- ошибки интерфейсов;
- ошибки объема данных и др.

Логические ошибки являются причиной нарушения логики алгоритма, внутренней несогласованности переменных и операторов, а также правил программирования. Функциональные ошибки - следствие неправильно определенных функций, нарушения порядка их применения или отсутствия полноты их реализации и т.д.

Ошибки вычислений возникают по причине неточности исходных данных и реализованных формул, погрешностей методов, неправильного применения операций вычислений или операндов. Ошибки времени выполнения связаны с необеспечением требуемой скорости обработки запросов или времени восстановления программы.

Ошибки ввода-вывода и манипулирования данными являются следствием некачественной подготовки данных для выполнения программы, сбоев при занесении их в базы данных или при выборке из нее.

Ошибки интерфейса относятся к ошибкам взаимосвязи отдельных элементов друг с другом, что проявляется при передаче данных между ними, а также при взаимодействии со средой функционирования.

Ошибки объема относятся к данным и являются следствием того, что реализованные методы доступа и размеры баз данных не удовлетворяют реальным объемам информации системы или интенсивности их обработки.

Приведенные основные классы ошибок свойственны разным типам компонентов ПО и проявляются они в программах по-разному. Так, при работе с БД возникают ошибки представления и манипулирования

данными, *логические ошибки* в задании прикладных процедур обработки данных и др. В программах вычислительного характера преобладают ошибки вычислений, а в программах управления и обработки - логические и функциональные ошибки. В ПО, которое состоит из множества разноплановых программ, реализующих разные функции, могут содержаться ошибки разных типов. Ошибки интерфейсов и нарушение объема характерны для любого типа систем.

Анализ типов ошибок в программах является необходимым условием создания планов тестирования и методов тестирования для обеспечения правильности ПО.

На современном этапе развития средств поддержки разработки ПО (*CASE-технологии*, объектно-ориентированные методы и средства проектирования моделей и программ) проводится такое проектирование, при котором ПО защищается от наиболее типичных ошибок и тем самым предотвращается появление программных дефектов.

Связь ошибки с отказом. Наличие ошибки в программе, как правило, приводит к отказу ПО при его функционировании. Для анализа причинно-следственных связей "ошибкаотказ" выполняются следующие действия:

- идентификация изъянов в технологиях проектирования и программирования;
- взаимосвязь изъянов процесса проектирования и допускаемых человеком ошибок;
- классификация отказов, изъянов и возможных ошибок, а также дефектов на каждом этапе разработки;- сопоставление ошибок человека, допускаемых на определенном процессе разработки, и дефектов в объекте, как следствий ошибок спецификации проекта, моделей программ;

- проверка и защита от ошибок на всех этапах ЖЦ, а также обнаружение дефектов на каждом этапе разработки;
- сопоставление дефектов и отказов в ПО для разработки системы взаимосвязей и методики локализации, сбора и анализа информации об отказах и дефектах;
- разработка подходов к процессам документирования и испытания ПО.

Конечная цель причинно-следственных связей "ошибкаотказ" заключается в определении методов и средств тестирования и обнаружения ошибок определенных классов, а также критериев завершения тестирования на множестве наборов данных; в определении путей совершенствования организации процесса разработки, тестирования и сопровождения ПО.

Приведем следующую классификацию типов отказов:

- аппаратный, при котором общесистемное ПО не работоспособно;
- информационный, вызванный ошибками во входных данных и передаче данных по каналам связи, а также при сбое устройств ввода (следствие аппаратных отказов);
- эргономический, вызванный ошибками оператора при его взаимодействии с машиной (этот отказ - вторичный отказ, может привести к информационному или функциональному отказам);
- программный, при наличии ошибок в компонентах и др.

Некоторые ошибки могут быть следствием недоработок при определении требований, проекта, генерации выходного кода или документации. С другой стороны, они порождаются в процессе разработки программы или при разработке интерфейсов отдельных элементов программы (нарушение порядка параметров, меньше или больше параметров и т.п.).

Источники ошибок. Ошибки могут быть порождены в процессе разработки проекта, компонентов, кода и документации. Как правило, они обнаруживаются при выполнении или сопровождении программного обеспечения в самых неожиданных и разных ее точках.

Некоторые ошибки в программе могут быть следствием недоработок при определении требований, проекта, генерации кода или документации. С другой стороны, ошибки порождаются в процессе разработки программы или интерфейсов ее элементов (например, при нарушении порядка задания параметров связи - меньше или больше, чем требуется и т.п.).

Причиной появления ошибок - непонимание требований заказчика; неточная спецификация требований в документах проекта и др. Это приводит к тому, что реализуются некоторые функции системы, которые будут работать не так, как предлагает заказчик. В связи с этим проводится совместное обсуждение заказчиком и разработчиком некоторых деталей требований для их уточнения.

Команда разработчиков системы может также изменить синтаксис и семантику описания системы. Однако некоторые ошибки могут быть не обнаружены (например, неправильно заданы индексы или значения переменных этих операторов).