

## Методы отладки

Отладка программы в любом случае предполагает обдумывание и логическое осмысление всей имеющейся информации об ошибке. Большинство ошибок можно обнаружить по косвенным признакам посредством тщательного анализа текстов программ и результатов тестирования без получения дополнительной информации. При этом используют различные методы отладки программ:

- обратного прослеживания;
- индукции;
- дедукции;
- ручного тестирования.

### Общая методика отладки программного обеспечения

*1 этап* – изучение проявления ошибки – если выдано какое-либо сообщение или выданы неправильные или неполные результаты, то необходимо их изучить и попытаться понять, какая ошибка могла так проявиться. При этом используют индуктивные и дедуктивные методы отладки. В результате выдвигают версии о характере ошибки, которые необходимо проверить. Для этого можно применить методы и средства получения дополнительной информации об ошибке. Если ошибка не найдена или система просто «зависла», переходят ко второму этапу.

*2 этап* – локализация ошибки – определение конкретного фрагмента, при выполнении которого произошло отклонение от предполагаемого вычислительного процесса.

Локализация может выполняться:

- путем отсечения частей программы, причем если при отсечении некоторой части программы ошибка пропадает, то это может означать как то, что ошибка связана с этой частью, так и то, что внесенное исправление изменило проявление ошибки;
- с использованием отладочных средств, позволяющих выполнить интересующий нас фрагмент программы в пошаговом режиме и получить дополнительную информацию о месте проявления и характере ошибки, например, уточнить содержимое указанных переменных.

При этом если были получены неправильные результаты, то в пошаговом режиме проверяют ключевые точки процесса формирования данного результата. Как подчеркивалось выше, ошибка не обязательно допущена в том месте, где она проявилась. Если в конкретном случае это так, то переходят к следующему этапу.

*3 этап* – определение причины ошибки – изучение результатов второго этапа и формирование версий возможных причин ошибки. Эти версии необходимо проверить, возможно, используя отладочные средства для просмотра последовательности операторов или значений переменных.

*4 этап* – исправление ошибки – внесение соответствующих изменений во все операторы, совместное выполнение которых привело к ошибке.

*5 этап* – повторное тестирование – повторение всех тестов с начала, так как при исправлении обнаруженных ошибок часто вносят в программу новые.

Следует более тщательно проверять фрагменты программного обеспечения, где уже были обнаружены ошибки, так как вероятность ошибок в этих местах по статистике выше. Это вызвано следующими причинами. Во-первых, ошибки чаще допускают в сложных местах или в тех случаях, если спецификации на реализуемые операции недостаточно проработаны. Во-вторых, ошибки могут быть результатом того, что программист устал, отвлекся или плохо себя чувствует. В-третьих, как уже упоминалось выше, ошибки часто появляются в результате исправления уже найденных ошибок.

**Метод обратного прослеживания** используется для небольших программ. Начинается проверка с точки вывода неправильного результата. Для этой точки строится гипотеза о значениях основных переменных, которые могли бы привести к получению имеющегося результата. Далее, исходя из этой гипотезы, делают предположения о значениях переменных в предыдущей точке. Процесс продолжают, пока не обнаружат причину ошибки.

**Метод индукции** основан на тщательном анализе симптомов ошибки, которые могут проявляться как неверные результаты вычислений или как сообщение об ошибке. Если компьютер просто «зависает», то фрагмент проявления ошибки вычисляют исходя из последних полученных результатов и действий пользователя. Полученную таким образом информацию можно изучить, просматривая соответствующий фрагмент программы. В результате выдвигаются гипотезы об ошибках, которые затем проверяются. Если гипотеза верна, то детализируют информацию об ошибке, иначе – выдвигают другую гипотезу. Собирая данные об ошибке, необходимо зафиксировать все, что известно об ее проявлениях, причем принимают во внимание, как ситуации, в которых фрагмент с ошибкой выполняется нормально, так и ситуации, в которых ошибка проявляется. Если в результате изучения данных никаких гипотез не появляется, то необходима дополнительная информация об ошибке.

Индуктивный анализ означает движение от частного к целому. При этом, просматривая симптомы ошибки и взаимосвязи между ними, часто можно прийти к ошибке.

Процесс индукции разбивается на следующие шаги:

- 1) Определение данных, имеющих отношение к ошибке.
- 2) Организация данных.
- 3) Выдвижение гипотезы.
- 4) Доказательство или опровержение гипотезы.

Основная ошибка, встречающаяся в плохо организованной отладке программы, заключается в том, что рассматриваются все имеющиеся данные. Первым шагом индуктивного метода должно быть перечисление всех действий, свидетельствующих о правильном выполнении программы и всех ее неправильных действий.

Второй шаг заключается в структурировании данных, имеющих отношение к ошибке, с целью выявления некоторых закономерностей. Особую важность имеет исследование на противоречивость, то есть, когда ошибка проявляется при правильных входных значениях и отсутствует при вводе неправильных.

Для изучения взаимосвязи между признаками ошибки выдвигаются гипотезы о ее причинах. Если гипотезы нельзя выдвинуть, то необходимы дополнительные данные (а для этого могут потребоваться дополнительные тесты).

Гипотеза доказывается путем сравнения ее с первоначальными симптомами ошибки или данными. Она должна полностью объяснить существование этих симптомов. Если такое объяснение получить не удастся, то или гипотеза не полна, либо не верна.



Схема процесса отладки методом индукции



Схема процесса отладки методом дедукции

**Метод дедукции** заключается в следующем. Сначала формируют множество причин, которые могли бы вызвать данное проявление ошибки. Затем, анализируя причины, исключают те, которые противоречат имеющимся данным. Если все причины исключены, то следует выполнить дополнительное тестирование исследуемого фрагмента. В противном случае наиболее вероятную гипотезу пытаются доказать. Если гипотеза объясняет полученные признаки ошибки, то ошибка найдена, иначе – проверяют следующую причину.

Процесс дедукции позволяет на основании некоторых общих теорий и предпосылок, используя операции исключения и уточнения, прийти к

определенному заключению, то есть обнаружить место ошибки. Он состоит из следующих этапов:

1. Перечисление возможных причин и гипотез.
2. Использование данных для исключения возможных причин.
3. Уточнение выбранной гипотезы.
4. Доказательство выбранной гипотезы.

Первый шаг заключается в разработке списка всех возможных причин ошибки. Эти причины могут являться только версиями, с помощью которых можно структурировать и анализировать имеющиеся в распоряжении данные. Далее, путем тщательного анализа данных исключаются все возможные причины, кроме одной. Если исключить все причины, то потребуются дополнительные данные (например, за счет дополнительных тестов) для выдвижения новых гипотез. Если же остается более чем одна причина, то первой выбирается наиболее вероятная из них.

Возможная причина может быть определена верно, но она может недостаточно полно отражать специфику ошибки. Поэтому следующим шагом должно быть использование доступных данных для уточнения гипотезы (например, ошибка при обращении к последней записи файла) с учетом некоторой специфики (например, последняя запись в буфере затирает признак конца файла).

**Метод ручного тестирования** – самый простой и естественный способ отладки программы. При обнаружении ошибки необходимо выполнить тестируемую программу вручную, используя тестовый набор, при работе с которыми была обнаружена ошибка. Метод эффективен, но не применим для программ со сложными вычислениями, для больших программ, а также в случаях, когда ошибка связана с неверным представлением программиста о выполнении операций. Данный метод часто используют как составную часть других методов отладки.