Errors Самойлов Савелий КИ-25

Програаммная ошиибка— означает ошибку в программе или в системе, из-за которой программа выдает неожиданное поведение и, как следствие, результат. Большинство программных ошибок возникают из-за ошибок, допущенных разработчиками программы в её исходном коде, либо в её дизайне. Также некоторые ошибки возникают из-за некорректной работы инструментов разработчика, например из-за компилятора, вырабатывающего некорректный код.

Термин «программная ошибка» обычно употребляется для обозначения ошибок, проявляющих себя на стадии работы программы, в отличие, например, от ошибок проектирования или синтаксических ошибок. Отчет, содержащий информацию об ошибке также называют отчетом о проблеме (англ. bug report). Отчет о критической проблеме (англ. crash), вызывающей аварийное завершение программы, называют крэш-репортом (англ. crash report).

Программные ошибки локализуются и устраняются в процессе тестирования и отладки программы.

В зависимости от этапа разработки ПО, на котором выявляется ошибка, выделяют:

синтаксические ошибки (распознаваемые в качестве таковых транслятором и делающие компиляцию невозможной) — например отсутствие или несоответствие открывающей и закрывающей скобок;

предупреждения (warnings) компилятора — например, использование неинициализированной переменной. В этом случае компилятор может заметить, что программист делает что-то необычное (вероятно неверное), и сообщает об этом, однако программист сам принимает решение, игнорировать сообщение или нет;

ошибки времени исполнения, смысловые ошибки (семантические) — например вычитание переменных вместо сложения или ошибка сегментации.

По важности:

Критические (иногда showstoppers) — лишающие программу всей полезности;

Серьёзные;

Незначительные;

Косметические.

По времени появления:

Постоянно, при каждом запуске;

Иногда («плавающий» тип);

Только на машине у клиента (зависит от локальных настроек у клиента).

По месту и направлению:

Ошибки пользовательского интерфейса;

Системы обработки ошибок;

Ошибки, связанные с граничными условиями (например, некорректная обработка пустой строки или максимального числового значения);

Ошибки вычислений;

Ошибки управления потоками;

Ошибки обработки или интерпретации данных;

При состоянии гонки;

Повышение нагрузки;

Ошибки контроля версии и идентификаторов;

Ошибки тестирования.

В зависимости от характера ошибки, программы и среды исполнения, ошибка может проявляться сразу или наоборот — долгое время оставаться незамеченной (например Проблема 2038 года).

Также ошибка может проявляться в виде уязвимости, делающей возможным несанкционированный доступ к системе или DoS-атаку.

Какие типичные ошибки в программировании совершают новички:

**Изобретают велосипед.**

**Склонность к «красивым» решениям.**

**Использование объектов не по назначению**

**Пренебрежение стилю написания кода,**

**Недостаточное или некорректное комментирование кода**

**Выводы:**

**1.**Syntax error. Неверное использование языка. Часто лишние или недостающие скобки или кавычки. Что делать? Заменить, удалить или добавить символы. Часто проблема в скобках или кавычках: открытые скобки должны быть закрыты, открытые кавычки должны быть закрыты.

**2**.Reference error. Использование несуществующего названия. Что делать? Проверить, существует ли то, на что вы ссылаетесь. Возможно вы использовали ошибочное название или забыли его создать.

**3**.Type error. Использование неверного типа, например попытка вызвать константу числа, как функцию. Что делать? Убедиться, что всё используется верно. Часто проблема в простой путанице: вы создали численную константу и функциональную константу, а потом пытаетесь вызвать число. Вероятно вы собирались вызвать функцию.

**4.**Logic error. (Логическая ошибка) Ваш код выполняет не то, что требуется, но программа запускается и не выдаёт ошибок трёх перечисленных выше типов. Сломана логика. Что делать? Проверить свой код, убедиться, что он выполняет то, что должен.