

## §7. Ошибки в программах

### 7.1. Характерные ошибки программирования

Ошибки могут быть допущены на всех этапах решения задачи – от ее постановки до оформления. Разновидности ошибок и соответствующие примеры приведены в таблице:

Вид ошибки	Пример
Неправильная постановка задачи	Правильное решение неверно сформулированной задачи
Неверный алгоритм	Выбор алгоритма, приводящего к неточному или эффективному решению задачи
Ошибка анализа	Неполный учет ситуаций, которые могут возникнуть; логические ошибки
Семантические ошибки	Непонимание порядка выполнения оператора
Синтаксические ошибки	Нарушение правил, определяемых языком программирования
Ошибки при выполнении операций	Слишком большое число, деление на ноль, извлечение квадратного корня из отрицательного числа и т.п.
Ошибки в данных	Неудачное определение возможного диапазона изменения данных
Опечатки	Перепутаны близкие по написанию символы, например, цифра 1 и буквы <i>I, l</i>
Ошибки ввода-вывода	Неверное считывание входных данных, неверное задание форматов данных

### 7.2. Синтаксические ошибки

Обычно синтаксические ошибки выявляются на этапе трансляции. Многие же другие ошибки транслятору выявить невозможно, так как транслятору неизвестны замыслы программиста.

Отсутствие сообщений машины о синтаксических ошибках является необходимым, но не достаточным условием, чтобы считать программу правильной.

Примеры синтаксических ошибок:

- пропуск знака пунктуации;
- несогласованность скобок;
- неправильное формирование оператора;
- неверное образование имен переменных;
- неверное написание служебных слов;

- отсутствие условий окончания цикла;
- отсутствие описания массива и т.п.

### **7.3. Ошибки не обнаруженные транслятором**

Существует множество ошибок, которые транслятор выявить не в состоянии, если используемые в программе операторы сформированы верно. Приведем примеры таких ошибок.

#### **Логические ошибки:**

- неверное указание ветви алгоритма после проверки некоторого условия;
- неполный учет возможных условий;
- пропуск в программе одного или более блоков алгоритма.

#### **Ошибки в циклах:**

- неправильное указание начала цикла;
- неправильное указание условий окончания цикла;
- неправильное указание числа повторений цикла;
- бесконечный цикл.

#### **Ошибки ввода-вывода; ошибки при работе с данными:**

- неправильное задание тип данных;
- организация считывания меньшего или большего объёма данных, чем требуется;
- неправильное редактирование данных.

#### **Ошибки в использовании переменных:**

- использование переменных без указания их начальных значений;
- ошибочное указание одной переменной вместо другой.

#### **Ошибки при работе с массивами:**

- массивы предварительно не обнулены;
- массивы неправильно описаны;
- индексы следуют в неправильном порядке.

#### **Ошибки в арифметических операциях:**

- неверное указание типа переменной (например, целочисленного вместо вещественного);
- неверное определение порядка действий;
- деление на нуль;
- извлечение квадратного корня из отрицательного числа;
- потеря значащих разрядов числа.

Все эти ошибки обнаруживаются с помощью тестирования.

#### **7.4. Сопровождение программы**

**Сопровождение программ** – это работы, связанные с обслуживанием программ в процессе их эксплуатации.

Многokrатное использование разработанной программы для решения различных задач заданного класса требует проведения следующих дополнительных работ:

- исправление обнаруженных ошибок;
- модификация программы для удовлетворения изменяющихся эксплуатационных требований;
- доработка программы для решения конкретных задач;
- проведение дополнительных тестовых просчетов;
- внесение исправлений в рабочую документацию;
- усовершенствование программы и т.д.

Применительно ко многим программам работы по сопровождению поглощают более половины затрат, приходящихся на весь период времени существования программы (начиная от выработки первоначальной концепции и кончая моральным ее устареванием) в стоимостном выражении.

Программа, предназначенная для длительной эксплуатации, должна иметь соответствующую документацию и инструкцию по её использованию.