

Обработка ошибок

среда, 8 декабря 2021 г. 11:47

- **Задача компилятора**
 - **Выявлять некорректные** программы
 - **Транслировать корректные** программы
- Существуют разные виды ошибок

Виды ошибок

Вид ошибка	Пример	Обнаруживает
Лексическая	...\$...	Лексер
Синтаксическая	...x*%...	Парсер
Семантическая	int x; y = x(3);	Контроль типов
Логическая	Любая программа	Тестировщик или пользователь

- **Простейший способ** - прерывать компиляцию при обнаружении первой ошибки
- **Плюсы**
 - Простая реализация
- **Минусы**
 - Показывают только одну ошибку
- **Обработчик ошибок парсера**
 - Должен ясно и точно **сообщать** о наличии ошибок (где и что за она)
 - Должен обеспечивать быстрое **восстановление** после ошибки
 - Столкнувшись с ошибкой компилятор должен не остановиться, а откатиться назад и подумать: как дальше все-таки продолжить, вернуться в состояние, с которого можно продолжить разбор
 - **Не должен** существенно **замедлять** компиляцию корректных программ

Методы восстановления ошибок

- **Восстановление в режиме паники**
 - Собрать информацию об ошибке, начать пропускать токены, пока не встретим один из синхронизирующих токенов
 - Самый простой и популярный метод
 - При обнаружении ошибки
 - Пропускать токены, пока не встретится токен определенного назначения
 - Продолжить с этого места
 - Основан на поиске синхронизирующих токенов
 - Обычно токены, завершающие операцию или выражение
 - Пример
 - (1 + + 2) + 3
 - Восстановление
 - Пропустить все токены до следующего целого числа
 - Пропустить все токены до закрывающей скобки
 - Bison: использовать специальный терминальный символ error для указания, сколько токенов пропускать
- **Восстановление на уровне фразы**
 - При обнаружении ошибки выполняется **локальная коррекция** входного потока
 - Притворяемся что всего хватает и продолжаем, но ошибку себе помечаем
 - Пример
 - Перед точкой с запятой, разделяющей операторы, автоматически закрывать все круглые скобки
- **Продукции ошибок**
 - Известные распространённые синтаксические ошибки включаются в грамматику
 - Пример
 - Для записи 5 x вместо 5*x
 - Включить в грамматику продукцию E --> ... | E E
 - **Недостаток**
 - Грамматика становится сложнее
- **Автоматическая локальная или глобальная коррекция**

- Идея: находить (подбирать) похожую (близкую) программу, не содержащую ошибок
 - Использовать вставку/удаление токенов
 - Использовать полный перебор
- Пример: PL/C - компилятор языка PL/1 с поддержкой автоматической коррекции
- Проблема
 - Может не угадать, что имел ввиду программист
 - Сложно будет отфиксировать логическую ошибку
- Прошлое
 - Длинный цикл компиляции/запуска программы (часы, дни)
 - Нужно находить за один цикл как можно больше ошибок
- Настоящее
 - Короткий цикл компиляции/запуска
 - Тенденция к исправлению одной ошибки за каждый цикл
 - Сложные механизмы обработки ошибок менее востребованы