

Д.В. РІСУХІН, В.І. НОСКОВ, д-р техн. наук, професор,
С.Ю. ГАВРИЛЕНКО, канд. техн. наук, професор

Дослідження основних принципів побудови трансляторів

Нині штучні мови, що використовують для опису предметної області текстового представлення, широко застосовуються не лише в програмуванні, але і в інших областях. З їх допомогою описується структура різних документів, трьохмірних віртуальних світів, графічних інтерфейсів користувача і багатьох інших об'єктів, які використовуються в моделях і у реальному світі. Для того, щоб ці текстові описи були коректно складені, а потім правильно розпізнані і інтерпретовані, використовуються спеціальні методи їх аналізу і перетворення. У основі методів лежить теорія мов і формальних граматик, а також теорія автоматів. Програмні системи, призначені для аналізу і інтерпретації текстів, називаються трансляторами. Методи організації процесів трансляції, що визначають реалізацію основних фаз, виконуються різними варіантами взаємодії блоків транслятора: лексичного аналізатора, синтаксичного аналізатора і генератора коду. Незважаючи на однаковий кінцевий результат, різні варіанти взаємодії блоків транслятора забезпечують різні варіанти зберігання проміжних даних.

При багатопрохідній організації взаємодії блоків трансляторів лексичний аналізатор повністю обробляє початковий текст, формуючи на виході ланцюжок, що складається з усіх отриманих лексем. Тільки після цього управління передається синтаксичному аналізатору. Синтаксичний аналізатор отримує сформований ланцюжок лексем і на її основі формує проміжне представлення або об'єктну модель. Після отримання усієї об'єктної моделі він передає управління генератору коду. Генератор коду, на основі об'єктної моделі мови, будує необхідний машинний код.

При однопрохідній організації взаємодії блоків транслятора найчастіше використовується схема управління, в якій роль основного блоку грає синтаксичний аналізатор. Лексичний аналізатор і генератор коду виступають в ролі підпрограм, що викликаються їм. Як тільки синтаксичному аналізатору потрібна чергова лексема, він викликає сканер. При отриманні фрагмента проміжного представлення здійснюється звернення до генератора коду. Завершення процесу трансляції відбувається після отримання і обробки останньої лексеми і ініціюється синтаксичним аналізатором.

Комбіновані взаємодії блоків транслятора – це поєднання багатопрохідної і однопрохідної схем трансляції, які породжують різноманітні комбіновані варіанти. Окрім цього, замість генератора коду легко підключити емулятор проміжного представлення, що дозволяє розробити систему програмування на деякій мові, орієнтовану на різні середовища виконання.

На фазі лексичного аналізу вхідна програма, що є потоком літер, розбивається на лексеми – слова відповідно до визначень мови. Основними формаліза-

мами, що лежать в основі реалізації лексичних аналізаторів, являються кінцеві автомати і регулярні вирази. Лексичний аналізатор може працювати в двох основних режимах: або як підпрограма, що викликається синтаксичним аналізатором для отримання чергової лексеми, або як повний прохід, результатом якого є файл лексем.

Тому актуальною задачею є дослідження основних принципів побудови трансляторів та пов'язаних з ними проблеми вибору найоптимальнішого методу реалізації, оскільки вже існує множина різних генераторів лексичного і синтаксичного аналізу, використання яких дозволяє скоротити час і підвищити ефективність.

В результаті виконаних в роботі досліджень отримані наступні наукові результати. Створена модель за методом побудови однопрохідного компілятора, що розділяє його реалізацію на графічну, текстову і таку, що інтерпретує частини і що дозволяє використати переваги кожної з них для спрощення побудови і супроводу компіляторів написаних вручну. Розроблено програмну модель використовуваних методів та вперше виконано їх порівняльний аналіз. Запропоновані в роботі методи дозволяють в короткі терміни розробити ефективний і надійний однопрохідний компілятор. Розроблені моделі для зручної побудови, аналізу, перетворення і використання методів трансляції для однопрохідного компілятора. Було проведено дослідження та аналіз розроблених методів побудови фаз транслятора, а також виявлення переваг та недоліків кожної з фаз враховуючи різні варіанти вхідних послідовностей на фазу лексичного аналізатора. Досліджено основні принципи побудови трансляторів, а саме побудова лексичного, синтаксичного, семантичного аналізаторів та методів оптимізації і генерації коду.

У роботі було досліджено основні науково-практичні проблеми принципів побудови трансляторів та пов'язаних з ними проблеми вибору найоптимальнішого методу на прикладі створення однопрохідного компілятора, оскільки вже існує множина різних генераторів лексичного і синтаксичного аналізу, використання яких дозволяє скоротити час і підвищити ефективність.

Список літератури:

1. Хантер Р. Основные концепции компиляторов / Р. Хантер. – Вильямс, 2002. – 256 с.
2. Harel D. Biting the silver bullet: Toward a brighter future for system development / D. Harel // Computer. – 1992. – Jan. – С. 8–20.
3. Гавриленко С.Ю. К вопросу о генерации кода в современных компиляторах / С.Ю. Гавриленко, П.С. Шитьков // Вестник НТУ "ХПИ". – Х.: НТУ "ХПИ", 2007. – С. 20–26.
4. Ульман Джеффри, Ахо Альфред, Сети Рави. Компиляторы: принципы, технологии и инструменты. – М: Вильямс, 2001. – 768 с.