Лабораторная работа №4 «Объектно-ориентированный лексический анализатор»

Скоробогатов С. Ю., Коновалов А. В.

22 марта 2016

1 Цель работы

Целью данной работы является приобретение навыка реализации лексического анализатора на объектно-ориентированном языке без применения каких-либо средств автоматизации решения задачи лексического анализа.

2 Задание

В лабораторной работе предлагается реализовать на языке Java две первые фазы стадии анализа: чтение входного потока и лексический анализ. При этом следует придерживаться схемы реализации объектно-ориентированного лексического анализатора, рассмотренной на лекции.

Входной поток должен загружаться из файла (в UTF-8). В результате работы программы в стандартный поток вывода должны выдаваться описания распознанных лексем в формате

```
Тег (координаты_фрагмента): атрибут лексемы
```

При этом для лексем, не имеющих атрибутов, нужно выводить только тег и координаты. Например,

```
IDENT (1, 2)-(1, 4): str
ASSIGN (1, 8)-(1, 9):
STRING (1, 11)-(1, 16): qwerty
```

Лексемы во входном файле могут разделяться пробельными символами (пробел, горизонтальная табуляция, маркеры окончания строки), а могут быть записаны слитно (если это не приводит к противоречиям).

Идентификаторы и числовые литералы не могут содержать внутри себя пробельных символов, если в задании явно не указано иного (варианты 3, 5 и 34). Комментарии, строковые и символьные литералы могут содержать внутри себя пробельные символы.

Лексический анализатор должен иметь программный интерфейс для взаимодействия с парсером. Рекомендуется реализовывать его как метод `nextToken()` для императивных языков или функцию, возвращающую список лексем, для функциональных языков.

Входной файл может содержать ошибки, при обнаружении которых лексический анализатор должен выдавать сообщение с указанием координаты, восстанавливаться и продолжать работу.

Для лексических доменов должны вычисляться их атрибуты:

- для целых чисел атрибут должен быть целым числом наибольшей разрядности (например, в Java — `long`).
- для вещественных чисел атрибут должен быть вещественным числом (например, `double` в Java или C++),
- для идентификаторов — номер в таблице идентификаторов (см. слайды лекции),
- для строковых констант — значение, изображаемое самой строковой константой (т.е. без окружающих кавычек и с интерпретацией escape-последовательностей),
- для комментариев токен не порождается, вместо этого координаты комментария помещается в список комментариев (см. слайды лекции).

Варианты языков для лексического анализа приведены в таблицах 1, 2, 3, 4 и 5.

Таблица 1: Краткое описание лексики вариантов языков (продолжение)

1	Географические координаты: начинаются с одного из знаков «S», «E», «N», «W», после которых располагается целое десятичное число, за которым может следовать либо точка и последовательность десятичных цифр, либо знак «D», за которым следует необязательная запись угловых минут (число от 0 до 59, за которым пишется апостроф) и угловых секунд (число от 0 до 59, за которым следует двойная кавычка). Атрибут лексемы (для лабораторных работ № 4 и № 6): вещественное число, соответствующее широте или долготе. Широта («S», «N») не может превышать 90, долгота («E», «W») — 180.
2	Числа: последовательности десятичных цифр. Знаки операций: «+», «-», «*», «/», «(», «)». Комментарии начинаются на «(*», заканчиваются на «*)», не могут быть вложенными.
3	Целочисленные переменные: начинаются с буквы «I», «J», «K», «L», «M» или «N», за которой может следовать не более пяти латинских букв и цифр. Вещественные переменные начинаются с любой другой буквы, за которой может следовать не более пяти латинских букв и цифр. Знаки: «+», «,». Целые числа — последовательности десятичных цифр. Пробелы и табуляции игнорируются (т.е. могут встречаться внутри переменных и чисел, не меняя их смысл).
4	Целые числа одинарной длины: последовательности десятичных цифр. Целые числа двойной длины: последовательности десятичных цифр с точкой на конце. Слова: любые последовательности непробельных символов, не являющиеся числами.
5	Открывающий тег: последовательность букв и цифр, окружённая «<» и «>». Закрывающий тег: последовательность букв и цифр, окружённая «</» и «>». Пробел (значимый токен): любая последовательность пробельных символов, Ключевое слово: «<», «>», «&», символ: любой печатный символ, кроме «<», «>» и «&». Для лабораторных работ № 4 и № 6: ключевые слова входят в домен символов и изображают символ соответствующей мнемоники HTML, атрибут тега — код в таблице идентификаторов (таблица идентификаторов общая для обоих видов тегов).
6	Целые числа: последовательности цифр определенной системы счисления, предваренные соответствующим индикатором, определяющим систему счисления (для десятичных чисел — пустой индикатор, для двоичных чисел — «0b», для восьмеричных чисел — «0t», для шестнадцатеричных чисел — «0x»). Ключевые слова: «and», «or». Знаки операций: «(», «)». Идентификаторы: последовательности латинских цифр.

Таблица 2: Краткое описание лексики вариантов языков (продолжение)

7	Идентификаторы: последовательности латинских букв и цифр, начинающиеся с буквы. Строковые константы — последовательности строковых секций, записанных слитно. Строковые секции: либо последовательность символов, ограниченных апострофами, апостроф внутри строки описывается как два апострофа подряд, не пересекают границы строк текста, либо знак «#», за которым следует десятичная константа (код символа). Пример строковой константы: «'hello'#10#13'world'» (эта строковая константа состоит из 4 строковых секций, однако является единым токеном).
8	Строковые литералы: ограничены двойными кавычками, не могут пересекать границы текста, содержат escape-последовательности «\n», «\"», «\t» и «\\». Числовые литералы: последовательности десятичных знаков и знаков «_», начинающиеся с цифры (прочерк не влияет на значение числа).
9	Идентификаторы: последовательности буквенных символов Unicode, цифр и дефисов, начинающиеся с заглавной буквы. Директивы: любой знак валюты, после которого следует непустая последовательность заглавных букв.
10	Целочисленные константы: последовательности десятичных цифр, предваряемые символом #. Имена переменных: последовательности десятичных цифр, начинающиеся со знаков «.» или «:», имена массивов: последовательности десятичных цифр, начинающиеся с «,» или «;». Ключевые слова «PLEASE», «DO», «FORGET».
11	Химические вещества: последовательности латинских букв и цифр, начинающиеся с заглавной буквы, при этом после цифры не может следовать строчная буква (атрибут: строка). Примеры: «CuSO4», «CH3CH2OH», «Fe2O3». Коэффициенты: последовательности десятичных цифр. Между коэффициентом и веществом пробел может отсутствовать. Операторы: «+», «->».
12	Числа фибоначчиевой системы счисления: последовательности знаков «0» и «1», причём две единицы не могут соседствовать друг с другом. Идентификаторы: последовательности латинских букв, в которых гласные и согласные чередуются.
13	Слова искусственного языка Токипона конструируются из слогов. Если слог находится в начале слова, то он может не содержать первую согласную, в остальных случаях слог — это согласная + гласная + опциональная n. n не ставится, если за ней идут n или m. Из всех вариантов слогов в токипоне запрещены слоги: ji, ti, wo, wu (как труднопроизносимые). Гласные: a, e, i, o, u. Согласные: j, k, l, m, n, p, s, t, w. Необходимо написать лексический анализатор, определяющий слова языка Токипона. Знаки препинания: «.», «,», «?», «!». Атрибуты слов (для лабораторных работ № 4 и № 6): коды в таблице идентификаторов.

Таблица 3: Краткое описание лексики вариантов языков (продолжение)

14	Идентификаторы: последовательности буквенных символов Unicode и цифр, начинающиеся с буквы.Числовые литералы: десятичные литералы представляют собой последовательности десятичных цифр, шестнадцатиричные — начинаются на десятичную цифру, содержат шестнадцатеричные цифры (в любом регистре) и заканчиваются символом «h». Ключевые слова «mov», «eax».
15	Числовые литералы: знак «0» либо последовательности знаков «1». Строковые литералы: регулярные строки — ограничены двойными кавычками, могут содержать escape-последовательности «\"», «\t», «\n», не пересекают границы строк текста; буквальные строки — начинаются на «@"», заканчиваются на двойную кавычку, пересекают границы строк текста, для включения двойной кавычки она удваивается.
16	Идентификаторы: последовательности десятичных цифр. Числовые литералы: римские цифры или ключевое слово «NIHIL» (представляет значение 0), не чувствительны к регистру.
17	Идентификаторы: последовательности буквенных символов Unicode и цифр, начинающиеся с буквы, не чувствительны к регистру. Целочисленные константы: десятичные — последовательности десятичных цифр, шестнадцатеричные — последовательности шестнадцатиричных цифр, начинающиеся на «&H». Ключевые слова — «PRINT», «GOTO», «GOSUB» без учёта регистра.
18	Идентификаторы переменных: последовательности буквенных символов Unicode и цифр, начинающиеся на знаки «$», «@», «%». Имена функций: последовательности буквенных символов Unicode и цифр, начинающиеся на букву. Ключевые слова «sub», «if», «unless».
19	Регулярные выражения: ограничены знаками «/», не могут пересекать границы текста, содержат escape-последовательности «\n», «\/», «\\». Строковые литералы: ограничены тремя кавычками («"""»), могут занимать несколько строчек текста, не могут содержать внутри более двух кавычек подряд.
20	Идентификаторы: последовательности латинских букв и цифр, начинающиеся с буквы. Имена переменных: начинаются с префикса «s», «t» или «e», после которого может располагаться одна буква, одна цифра или точка («.»), за которой следует непустая последовательность латинских букв и цифр. Примеры имён переменных: «s1», «tx», «e.FileName», «t.666». (Домен имён переменных имеет приоритет.) Атрибуты (для лабораторных работ № 4 и № 6) идентификаторов и переменных — коды в таблицах. Таблицы идентификаторов и переменных различны.
21	Идентификаторы: последовательности буквенных символов Unicode и цифр, которые начинаются с буквы и могут заканчиваться на один из знаков «%», «$», «#», «&» или «!». Ключевые слова FOR, NEXT. Комментарии — любой текст, следующий за ключевым словом «REM» и продолжающийся до конца строки (то есть после буквы «M» в «REM» не может следовать буква или цифра). Операции: «+», «-», «/», «\». Идентификаторы и ключевые не чувствительны к регистру.

Таблица 4: Краткое описание лексики вариантов языков (продолжение)

22	Строковые литералы: органичены двойными кавычками, для включения двойной кавычки она удваивается, для продолжения литерала на следующей строчке текста в конце текущей строчки ставится знак «\». Числовые литералы: либо последовательности десятичных цифр, либо последовательности шестнадцатеричных цифр, начинающиеся с «$». Идентификаторы: последовательности буквенных символов Unicode, цифр и знаков «$», начинающиеся с буквы.
23	Идентификаторы: последовательности заглавных латинских букв, за которыми могут располагаться последовательности знаков «+», «-» и «*». Числовые литералы: знак «*» или последовательности, состоящие целиком либо из знаков «+», либо из знаков «-». Ключевые слова: «ON», «OFF», «**».
24	Комментарии: начинаются с «(*» или «{», заканчиваются на «*)» или «}» и могут пересекать границы строк текста. Целочисленные литералы: последовательности десятичных цифр. Дробные литералы: строки вида «digits/digits», где «digits» — последовательность десятичных цифр. Атрибут (для лабораторных работ № 4 и № 6) дробного числа — пара целых чисел (числитель и знаменатель).
25	Строковые литералы: ограничены обратными кавычками, могут занимать несколько строчек текста, для включения обратной кавычки она удваивается. Числовые литералы: десятичные литералы представляют собой последовательности десятичных цифр, двоичные — последовательности нулей и единиц, оканчивающиеся буквой «b». Идентификаторы: последовательности десятичных цифр и знаков «?», «*» и «|», не начинающиеся с цифры.
26	Идентификаторы: последовательности буквенных символов Unicode и десятичных цифр, начинающиеся и заканчивающиеся на букву. Числовые литералы: последовательности шестнадцатеричных цифр (чтобы литерал не был похож на идентификатор, его можно предварять нулём). Ключевые слова: «qeq», «xx», «xxx».
27	Символьные литералы: ограничены апострофами, могут содержать Escape-последовательности «\'», «\n», «\\» и «\xxxx» (здесь буквы «x» обозначают шестнадцатеричные цифры). Идентификаторы: последовательности буквенно-цифровых символов Unicode длиной от 2 до 10 символов, начинающиеся и заканчивающиеся буквой. Ключевые слова: «z», «for», «forward».
28	Идентификаторы: последовательности латинских букв и цифр, начинающиеся с буквы. Знаки операций: либо последовательности, состоящие из знаков !, #, $, %, &, *, +, ., /, <, =, >, ?, @, \, ^, |, - и ~, либо идентификаторы, записанные в обратных кавычках (например, «`plus`»). Ключевые слова «where», «->», «=>».
29	Комментарии: начинаются с «--» и продолжаются до конца строки, либо ограничены «{-» и «-}», могут занимать несколько строк текста. Целочисленные литералы: последовательности десятичных цифр. Вещественные литералы: последовательности десятичных цифр, за которой следует либо дробная часть (десятичная точка и последовательность десятичных цифр, возможно, пустая), либо показатель степени (буква «e» или «E», за которой следует не менее одной десятичной цифры), либо дробная часть и показатель степени.

Таблица 5: Краткое описание лексики вариантов языков

30	Идентификаторы: последовательности латинских букв, начинающиеся с гласной буквы. Числовые литералы: последовательности десятичных цифр, перед которыми может стоять знак «минус». Операции: «`--`», «`<`», «`<=`».
31	Комментарии: начинаются с «`/*`», заканчиваются на «`*/`» и могут пересекать границы строк текста. Идентификаторы: последовательности латинских букв и десятичных цифр, в которых буквы и цифры чередуются. Ключевые слова: «`for`», «`if`», «`m1`».
32	Строковые литералы: ограничены апострофами, для включения апострофа в литерал он удваивается, не пересекают границы строк текста. Числовые литералы: последовательности десятичных цифр, которые могут включать точку и предваряться знаком «минус». Идентификаторы: последовательности буквенных символов Unicode, точек и цифр, начинающиеся с буквы.
33	Идентификаторы: либо последовательности латинских букв, либо непустые последовательности десятичных цифр, ограниченные круглыми скобками. Числовые литералы: либо последовательности десятичных цифр, не начинающиеся с нуля, либо «`0`». Операции: «`()`», «`:`», «`:=`».
34	Комментарии: начинаются с «`//`» и продолжаются до окончания строки текста. Идентификаторы: любой текст, не содержащий «`/`» и ограниченный символами «`/`». Ключевые слова: «`/while/`», «`/do/`», «`/end/`».
35	Строковые литералы: ограничены двойными кавычками, могут содержать Escape-последовательности «`\"`», «`\n`», «`\t`» и «`\\`», не пересекают границы строк текста. Числовые литералы: последовательности десятичных цифр, разбитые точками на группы по три цифры («`100`», «`1.000`», «`1.000.000`»). Идентификаторы: последовательности латинских букв, знаков подчёркивания и цифр, начинающиеся с буквы или подчёркивания.
36	Идентификаторы: последовательности латинских букв и десятичных цифр, оканчивающиеся на цифру. Числовые литералы: непустые последовательности десятичных цифр, органиченные знаками «`<`» и «`>`». Операции: «`<=`», «`=`», «`==`».
37	Комментарии: целиком строка текста, начинающаяся с «`*`». Идентификаторы: либо последовательности латинских букв нечётной длины, либо последовательности символов «`*`». Ключевые слова: «`with`», «`end`», «`**`».