BİL 481 - Spring Homework 2

Prepared by Çağdaş Gerede

Due: April 20, 2021, 23:59
Submission: Send the following two files to *TA*
- A pdf file that has your ***handwritten*** answers for Q1 and Q2
- Your Runner.java file for Q3

Question 1:

When "java org.antlr.v4.gui.TestRig Expr s -tokens -input input.txt" program runs, it produces the following output. Write down the content of the input.txt file in the provided table below.
- Each table cell must contain a single character or no character.
- For table cells with no character, fill them with an X:

X

- For a new line character, fill the table with two characters as follows:

\r	\n

  (In Windows, a new line is \r\n. In Linux, \n. In MacOS, \r. Here, we assume we run the command in Windows)
- For a space character, use the $ character:

$

- For the end of file character, use EOF:

EOF

Output	Expr.g4

```[@0,0:0='5',<INT>,1:0]```
```[@1,2:2='+',<'+'>,1:2]```
```[@2,4:5='10',<INT>,1:4]```
```[@3,12:12='-',<'-'>,2:4]```
```[@4,14:15='18',<INT>,2:6]```
```[@5,32:32='+',<'+'>,5:9]```
```[@6,34:34='5',<INT>,5:11]```
```[@7,44:44='-',<'-'>,9:1]```
```[@8,46:46='2',<INT>,9:3]```
	```grammar Expr;```
	```s: e;```
	```e: e op=('*'|'/') e #MulDiv```
	``
		e op=('+'|'-') e #AddSub```
		``
			INT #Int;```
			```INT: [0-9]+;```
			```ADD: '+';```
			```SUB: '-';```
			```MUL: '*';```

```
[@9,48:48='*',<'*'>,9:5]
[@10,50:51='15',<INT>,9:7]
[@11,59:59='/',<'/'>,11:1]
[@12,61:61='5',<INT>,11:3]
[@13,62:61='<EOF>',<EOF>,11:
4]
```

```
DIV: '/';
WS: [ \r\n]+ -> skip;
```

	Col 0	Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9	Col 10	Col 11	Col 12	Col 13
Line 1														
Line 2														
Line 3														
Line 4														
Line 5														
Line 6														
Line 7														
Line 8														
Line 9														
Line 10														
Line 11														

Question 2:

Write a grammar file Cards.g4 that recognizes credit cards where each line consists of
- The local bank name,
- The card number whose card network will be categorized as Mastercard, Visa or Amex
- The expiration month and year,
- 3-digit CCV number, an optional comment.

The following command should produce the expected output below.

BİL 481 - Spring Homework 2

Prepared by Çağdaş Gerede

Command to run

```
java org.antlr.v4.Tool Cards.g4 & javac *.java & java org.antlr.v4.gui.TestRig Cards cards
-tokens input-bank.txt
```

---

input-bank.txt

```
Garanti 5120410012000120 21/01 130 /* Verified */
Garanti 5221512313500122 23/04 411 /* Second card*/
Garanti 5362710014050125 20/08 304 /* Inactive */
Garanti 5493990021000123 22/05 433
Garanti 5588803018014121 23/01 102
YapiKredi 4122410019105120 24/09 101
IS 3451813418014121 20/02 991 /* Advantage Plus */
```

Expected output

```
[@0,0:6='Garanti',<BANK>,1:0]
[@1,8:23='5120410012000120',<MASTERCARD>,1:8]
[@2,25:29='21/01',<EXPIRATION>,1:25]
[@3,31:33='130',<CCV>,1:31]
[@4,35:48='/* Verified */',<COMMENT>,channel=1,1:35]
[@5,51:57='Garanti',<BANK>,2:0]
[@6,59:74='5221512313500122',<MASTERCARD>,2:8]
[@7,76:80='23/04',<EXPIRATION>,2:25]
[@8,82:84='411',<CCV>,2:31]
[@9,86:101='/* Second card*/',<COMMENT>,channel=1,2:35]
[@10,104:110='Garanti',<BANK>,3:0]
[@11,112:127='5362710014050125',<MASTERCARD>,3:8]
[@12,129:133='20/08',<EXPIRATION>,3:25]
[@13,135:137='304',<CCV>,3:31]
[@14,139:152='/* Inactive */',<COMMENT>,channel=1,3:35]
[@15,155:161='Garanti',<BANK>,4:0]
[@16,163:178='5493990021000123',<MASTERCARD>,4:8]
[@17,180:184='22/05',<EXPIRATION>,4:25]
```

```
[@18,186:188='433',<CCV>,4:31]
[@19,191:197='Garanti',<BANK>,5:0]
[@20,199:214='5588803018014121',<MASTERCARD>,5:8]
[@21,216:220='23/01',<EXPIRATION>,5:25]
[@22,222:224='102',<CCV>,5:31]
[@23,227:235='YapiKredi',<BANK>,6:0]
[@24,237:252='4122410019105120',<VISA>,6:10]
[@25,254:258='24/09',<EXPIRATION>,6:27]
[@26,260:262='101',<CCV>,6:33]
[@27,265:266='IS',<BANK>,7:0]
[@28,268:283='3451813418014121',<AMEX>,7:3]
[@29,285:289='20/02',<EXPIRATION>,7:20]
[@30,291:293='991',<CCV>,7:26]
[@31,295:314='/* Advantage Plus */',<COMMENT>,channel=1,7:30]
[@32,315:314='<EOF>',<EOF>,7:50]
```

Notes:
- Credit cards are 16 digit numbers.
- Mastercard cards start with 51, 52, 53, 54, 55.
- Visa cards start with 4.
- American Express cards start with 34 or 37.
- Expiration is two digit year and two digit month separated by a '/', e.g., 20/09 (20/09 means: September 2020.)
- CCV is a 3-digit number.

Question 3:

Implement a Runner.java file similar to the ones from the lectures that when executed with a Java file it returns:
A. a list of methods that are declared but never called;
B. a list of methods that are called but never declared.

For example, given the following file, the tool must produce the output on the right:

Input:	Output:
`package com.acme;`	`Never called` `com.acme::B::m3`

```
public class A {                           com.acme::C::m1
      static void m1() {
              m2();                        Called but not declared
              B.m2();                      com.acme::A::m3
      }                                    com.acme::C::m3

      static void m2() {
              B.m1();
              m3();
      }
}

class B {
      static void m1() {
              A.m1();
      }

      static void m2() {
              C.m3();
      }

      static void m3() {
      }
}

class C {
      static void m1() {}
}
```

Each method output line is in the following format

<package name>::<class name>::<method name>

Assume that all the methods are static. There can be multiple classes in the input. We also assume that all the calls are within the input file (i.e., there is no call to a method declared outside the input file). On the other hand, there can be methods that are called but they are not declared yet (for instance, the call to C.m3).

The calls (such as *Math.abs*) to the methods that automatically come with the default java packages can also be assumed not declared. Also, you can assume there are no import statements in the input file.

BİL 481 - Spring Homework 2

Prepared by Çağdaş Gerede

For Java, use the grammar Java.g4 given here:

https://github.com/cagdasgerede/parsingWithAntlr/blob/master/11-AnnotatingMethodNames/Java.g4