Nick Krawczeniuk & Gianna Sorrentino

Grammars

Due on 11/13

The work has to be done alone or in a group of 2 students.

A hard copy is required. Professional presentation is important.

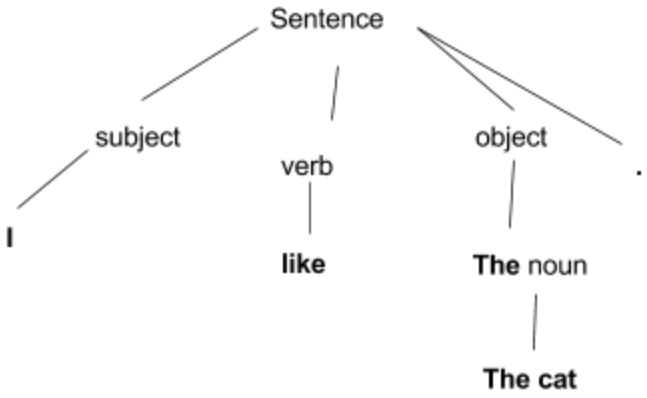
100 points

Exercise 1:

We consider the BNF grammar below:

Sentence	::=	Subject Verb Object .
Subject	::=	**I** | **a** Noun | **the** Noun
Object	::=	**me** | **a** Noun | **the** Noun
Noun	::=	**cat** | **mat** | **rat**
Verb	::=	**like** | **is** | **see** | **sees**

a. Show that **I like the cat.** is recognized by this BNF grammar using a rightmost derivation and, then, a parse tree.



Sentence → Subject Verb Object . → Subject Verb the noun . → Subject Verb the cat . → Subject like the cat . → I like the cat .

b. Provide an expression that is NOT recognized by the grammar.

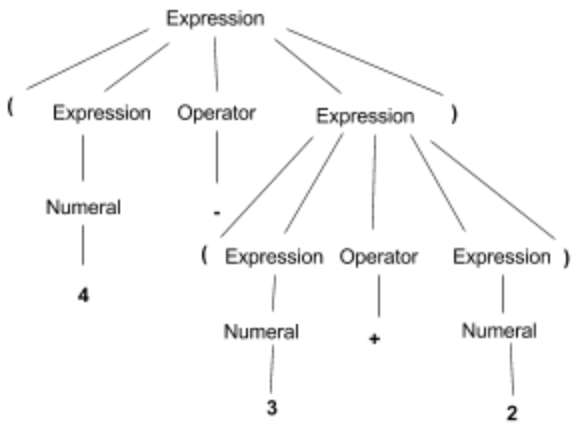
    a. She hates my dog

Exercise 2:

We consider the following grammar:

```
EXPRESSION ::= NUMERAL  | ( EXPRESSION OPERATOR EXPRESSION
)
NUMERAL ::=  0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
OPERATORS ::=  +  |  -
```

Show that (4 - (3 + 2)) is a legal EXPRESSION using a leftmost derivation, and,then, a parse tree.

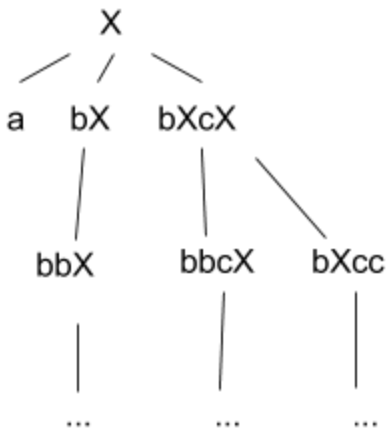


Expression → **(** Expression Operator Expression **)** → **(** Numeral Operator Expression **)** → **(4** Operator Expression**)** → **(4 -** Expression **)** → **(4 - (** Expression Operator Expression **))** → **(4 - (** Numeral Operator Expression **))** → **(4 - (3** Operator Expression **))** → **(4 - (3 +** Expression **))** → **(4 - (3 +** Numeral **))** → **(4 - (3 + 2))**

Exercise 3:

Show that the following grammar is ambiguous:

```
X -> a | bX | bXcX
where a,b,c are terminals.
```



This is ambiguous because you can have more than a single parse tree with bXcX. bX and bXcX are also essentially infinite because there are no cases defined for only "b" or "c" as there are with "a".

Exercise 4:

a. Design a BNF grammar that recognizes expressions of the form Ai where A is in {a,b,c} and i is a digit.

List → Ai
A → a | b | c
i → Digit

b. Design a BNF grammar that recognizes lists of the form A1, A2, A3, …, An. Use question a).

List → Ai
A → a | b | c
i → Digit | i Digit

Digit → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Exercise 5:

1. Write a JAY program that computes the sum of the *n* first numbers with a loop.

```
void main(){
int x;
int n;
int result;
x = 1;

if ( x <= n) {
        result = result + x;
        x = x + 1;
}
}
```

2. Write a JAY program that assigns the minimum of two numbers in a variable called min.

```
void main() {

int a;
int b;

if (a < b) {

min = a;

}
else {

min = b;

}
```

3. Provide 2 examples of lexical errors in JAY.
    a. A lexical error could be inputting an unaccepted character such as & or _.
4. Provide 2 examples of JAY programs with 2 different syntax errors.

```
void main() {

int boolean x;

}
```

```
void main() {

ifelse (x < y) {

  min = x;

  }

}
```

5. Provide 2 examples of JAY programs with errors that are neither detected during the lexical analysis nor during the syntactic analysis.

```
void main(){

x <=< 4;

}
```

```
void main(){

boolean x;

x + 2;

}
```