Kieran Finnegan

Problem Set One: BNF

Learning Abstract: In this problem set, English descriptions along with some examples were given. Then a BNF had to be created that followed the rules of the given languages. I learned how to take these descriptions and create then test a BNF.

Task 1: Backus-Naur Form is used to define the syntax of a language. It shows the rules of the language. It is written using non-terminals which are not part of the actual languages but are used to help convey the rules of a language. The tokens of the BNF are what is part of the language.

Task 2:

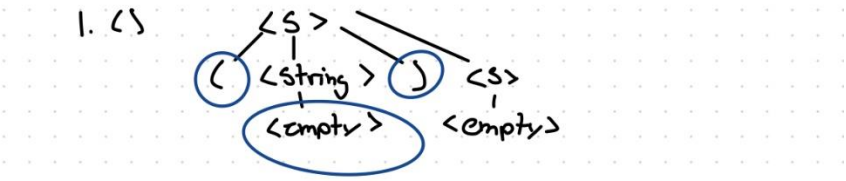
<PSL1> ::= (<string>) <PSL1> | <empty>

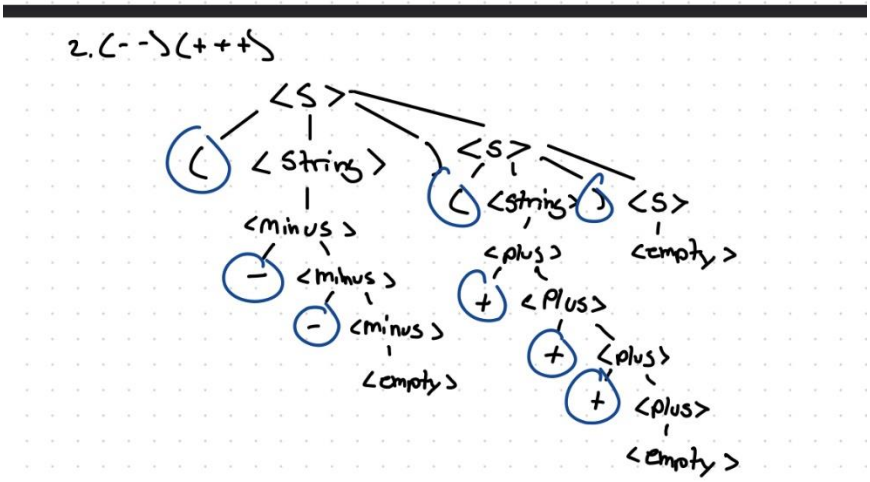
<string> :: = <plus> | <minus>

<plus> ::= +<plus> | <empty>

<minus> ::= -<minus> | <empty>

Task 3:





Task 4:

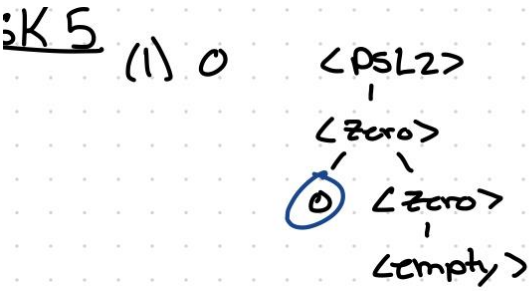
<PSL2> ::= <zero> | <non-zero>

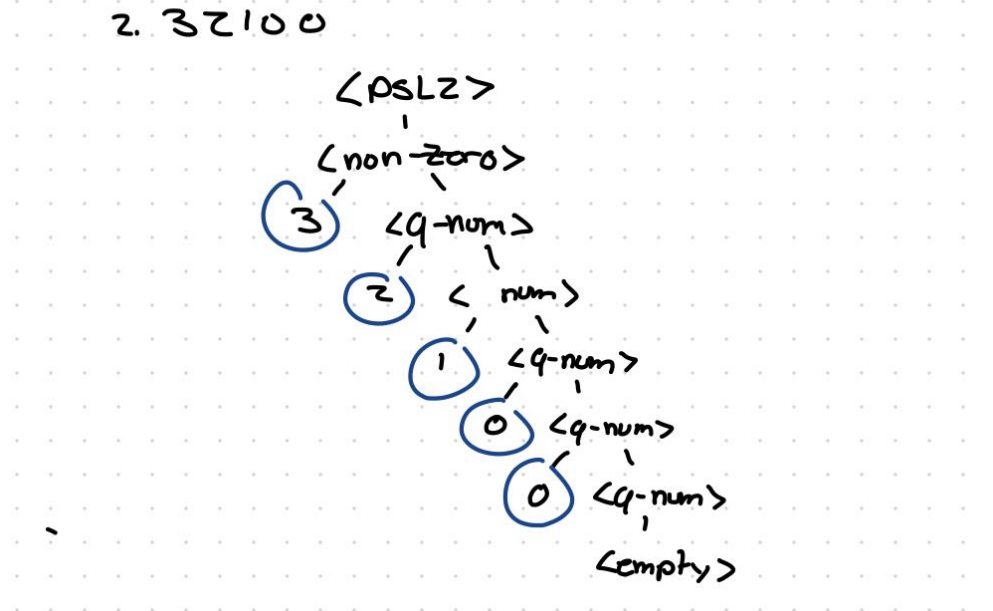
<zero> :: = 0<zero>| <empty>

<non-zero> ::= 1<q-num> | 2<q-num> | 3<q-num>

<q-num> ::= 0<q-num> | 1<q-num> | 2<q-num> | 3<q-num> | <empty>

Task 5:





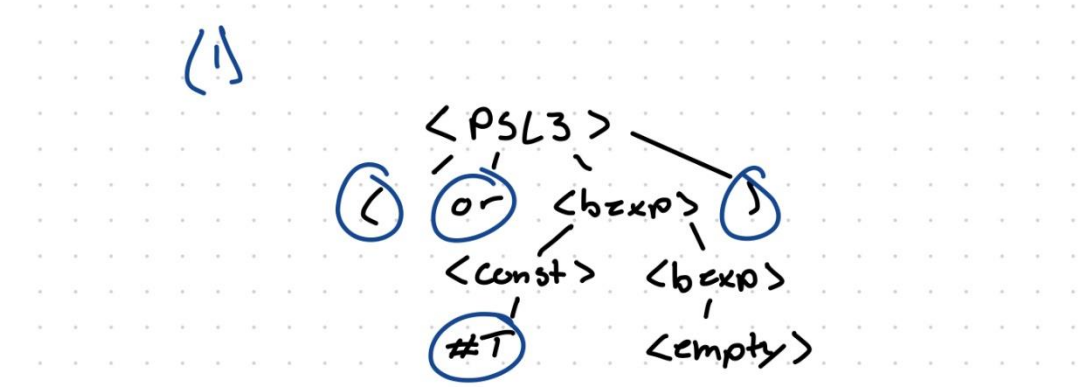
Task 6:

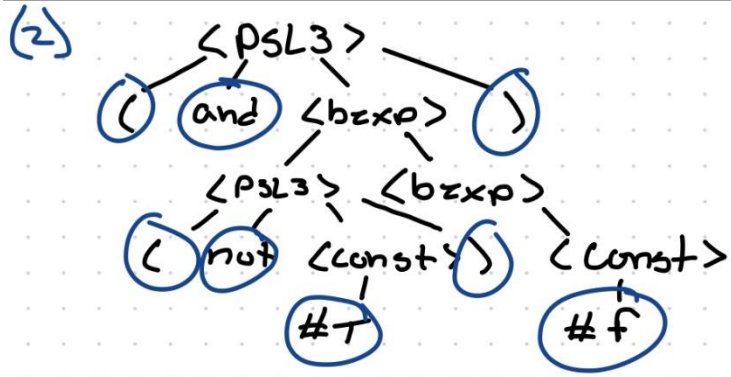
<PSL3> ::= (and <bexp>) | (or < bexp>) | (not <const>)

<bexp> ::= <const><bexp> | <PSL3> <bexp> | <empty>

<const> ::= #T | #F

Task 7:





Task 8:

<PSL4> ::= <hundreds> | <tens> | <teens> | <ones> | <zero>

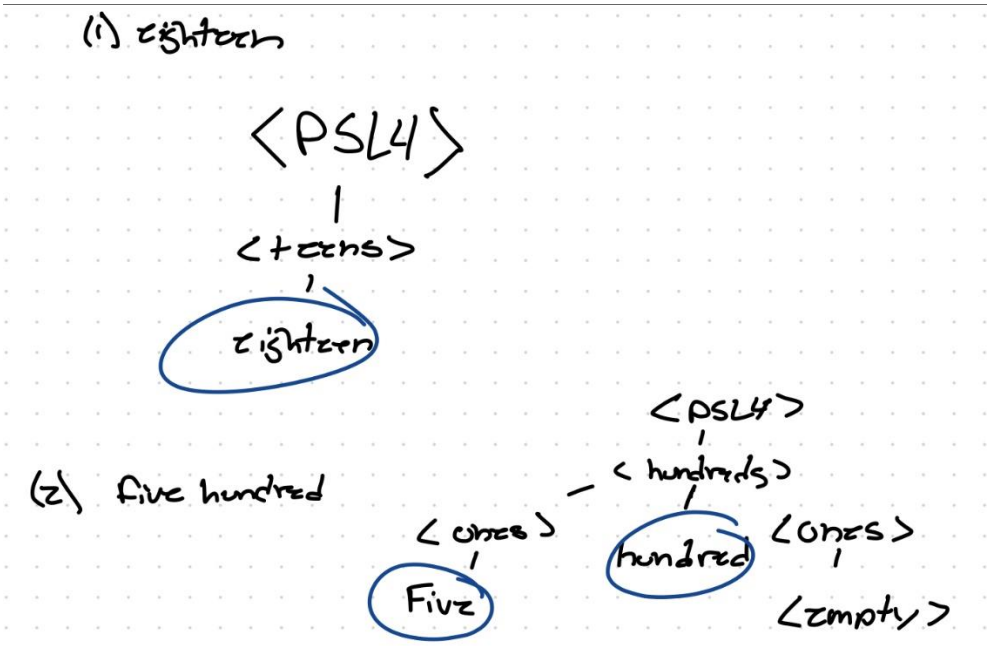
<hundreds> ::= <ones> hundred <tens> | <ones> hundred <teens> | <ones> hundred <ones>

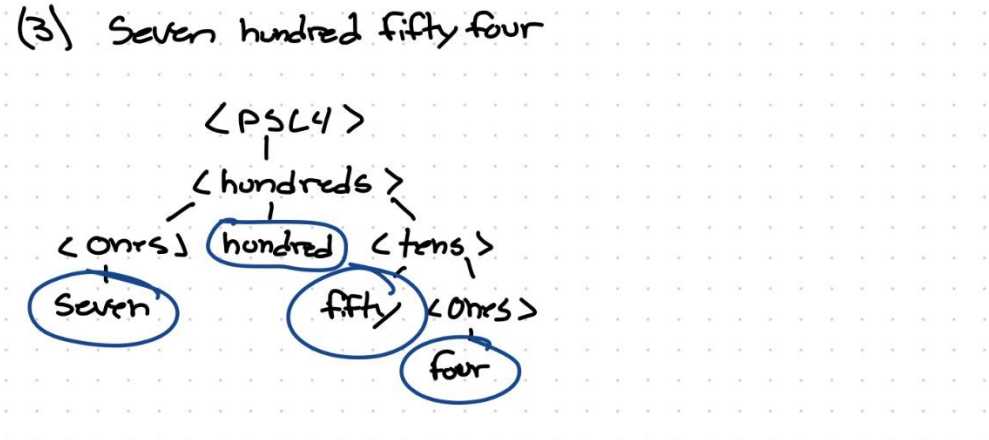
<tens :: = twenty <ones> | thirty <ones> | forty <ones> | fifty <ones> | sixty <ones> | seventy <ones> | eighty <one> | ninety <ones>

<teens> ::= eleven | twelve | thirteen | fourteen | fifteen | sixteen | seventeen | eighteen | nineteen

<ones> ::= one | two | three | four | five | six | seven | eight | nine | <empty>

Task 9:





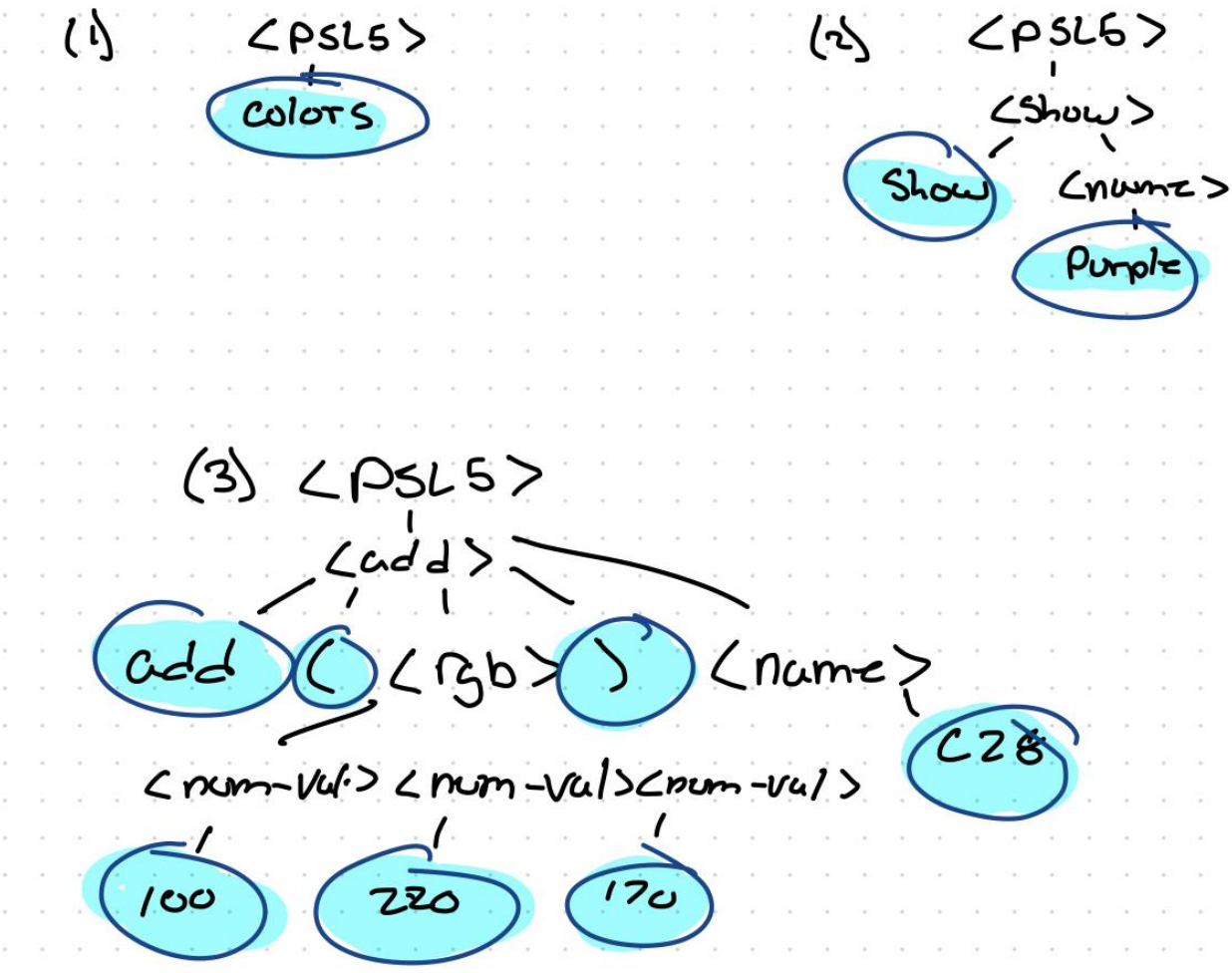
Task 10:

<PSL5> ::= <add> | <show> | <describe> | colors | exit

<add> ::= add (<rgb>) <name> | add color <name>

<show> ::= show <name>

<describe> ::= describe <name>

<rgb> ::= <num-value><num-value><num-value>

Task 11: