Problem Set Assignment #1: BNF Assignment

Learning Abstract

For this assignment, it is my first problem set assignment which focuses on BNF. Some BNF grammars for given languages and BNF parse trees based on these grammars were created. Moreover, BNF is described in natural language for this as well as its nature and significance.

Problem 1- Shapes

BNF Grammar description of the Shapes language

<Shapes> : : = (<size> <color> <pattern> <shape>)

<size> : : = (size large) | (size medium) | (size small)

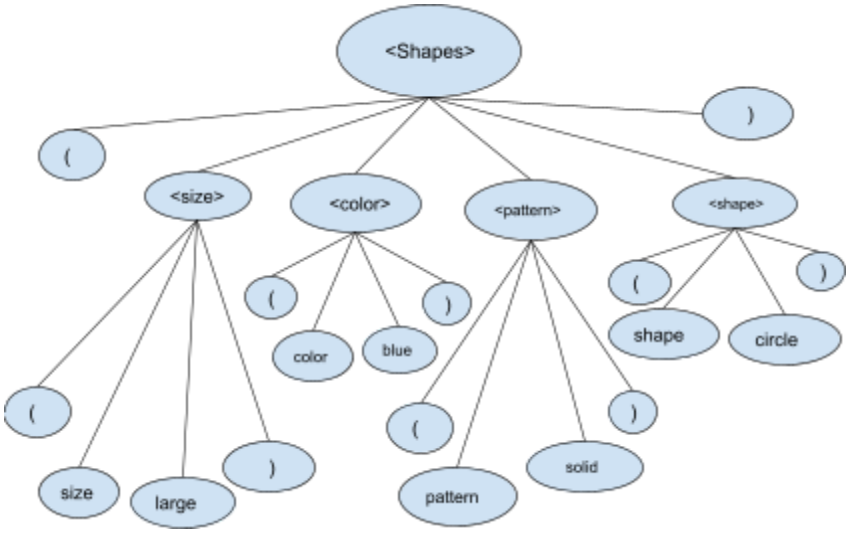
<color> : : = (color red) | (color blue) | (color yellow)

<pattern> : : = (pattern striped) | (pattern dotted) | (pattern solid)

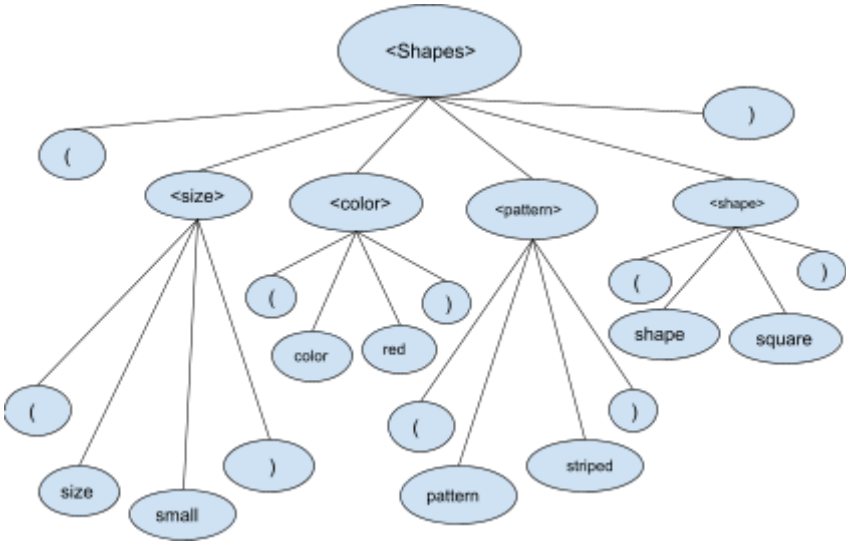
<shapes> : : = (shape circle) | (shape square) | (shape triangle)

BNF Parse Trees of the Shapes language

1. ((size big) (color blue) (pattern solid) (shape circle))



2. ((size small) (color red) (pattern striped) (shape square))



Problem 2 - SQN (Special Quaternary Numbers)

BNF Grammar description of the SQN language

<SQN> : : = 0 | 0 <non-zero> | <non-zero>

<non-zero> : : = <empty> | 1 <non-one> | 2 <non-two> | 3 <non-three>

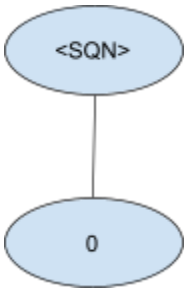
<non-one> : : = <empty> | 0 <non-zero> | 2 <non-two> | 3 <non-three>

<non-two> : : = <empty> | 0 <non-zero> | 1 <non-one> | 3 <non-three>

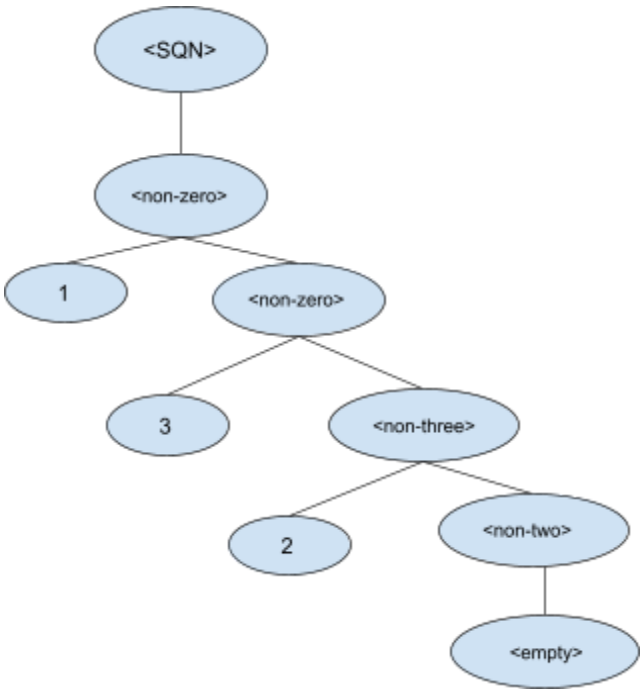
<non-three> : : = <empty> | 0 <non-zero> | 1 <non-one> | 2 <non-two>

BNF Parse Trees of the SQN language

1. 0



2. 132



3. The string, 1223, cannot be drawn as a parse tree in view of the fact that the specification based on the BNF grammar description of the SQN language does not allow for repeating digits.

Problem 3 - Fours

BNF Grammar description of the Fours language

<Fours> : : = <F1> | <F2> | <F3> | <F4>

<F1> : : = (1 1 1 1) | (1 1 1 1) <F2> <F3> <F4>

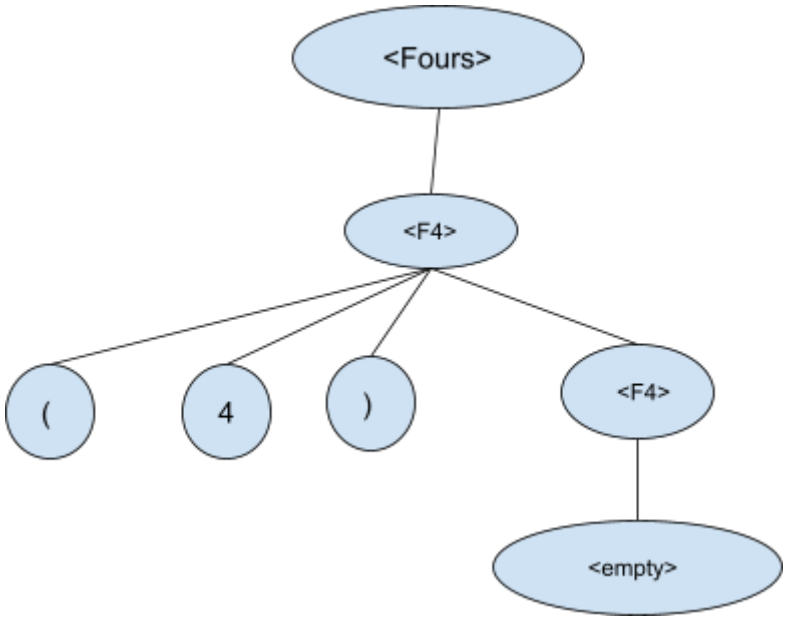
<F2> : : = <empty> | (1 1 2) <F2> | (1 2 1) <F2> | (2 1 1) <F2>

<F3> : : = <empty> | (3 1) <F3> | (1 3) <F3>

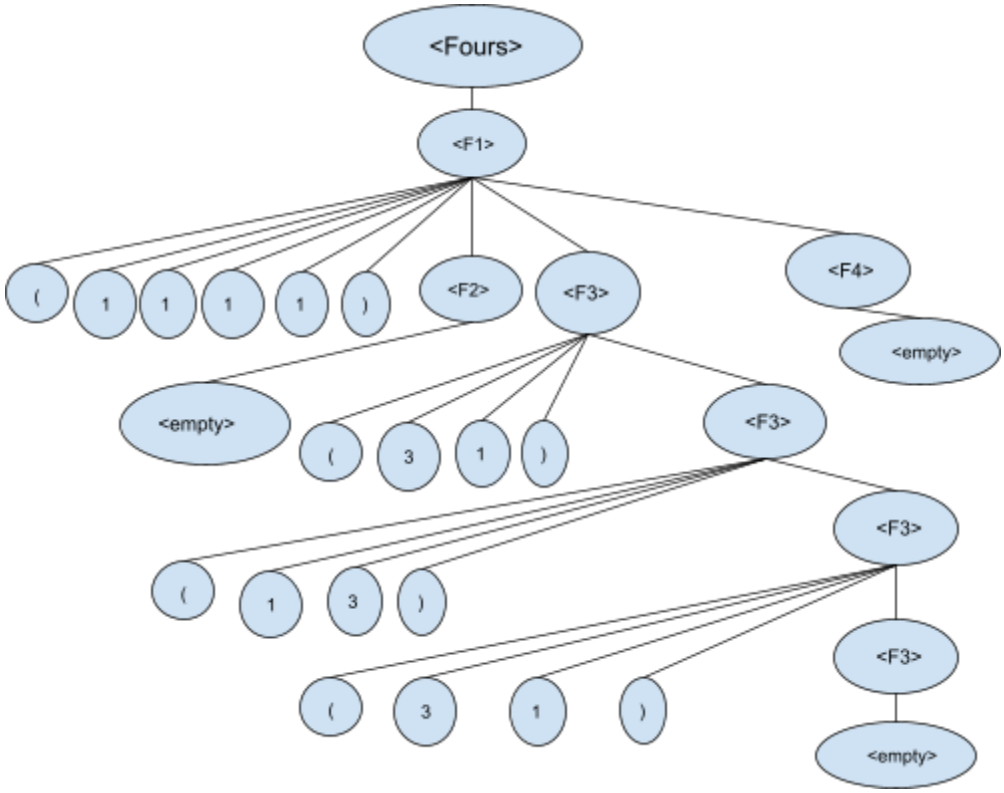
<F4> : : = <empty> | (4) <F4>

BNF Parse Trees of the Fours language

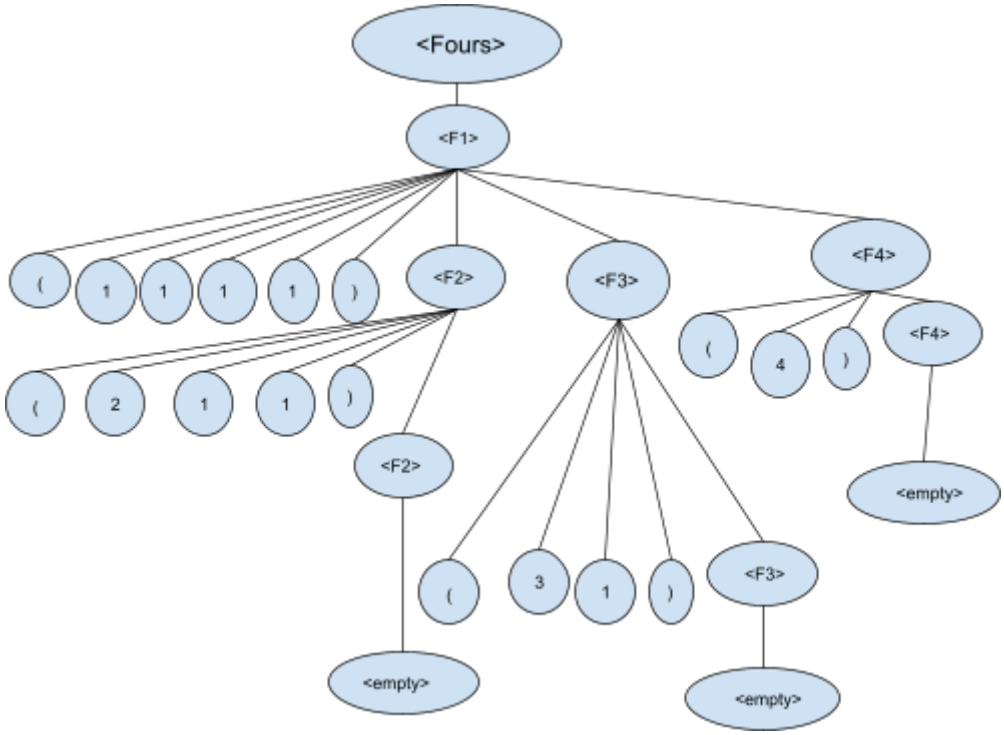
1. (4)



2. (1 1 1 1) (3 1) (1 3) (3 1)



3. (1 1 1 1) (2 1 1) (3 1) (4)



Problem 4 - BXR

BNF Grammar description of the BXR language

<BXR> : : = <operator>

<operator> : : = <empty>

| (and <operator> <constants> <operator>)

| (or <operator> <constants> <operator>)

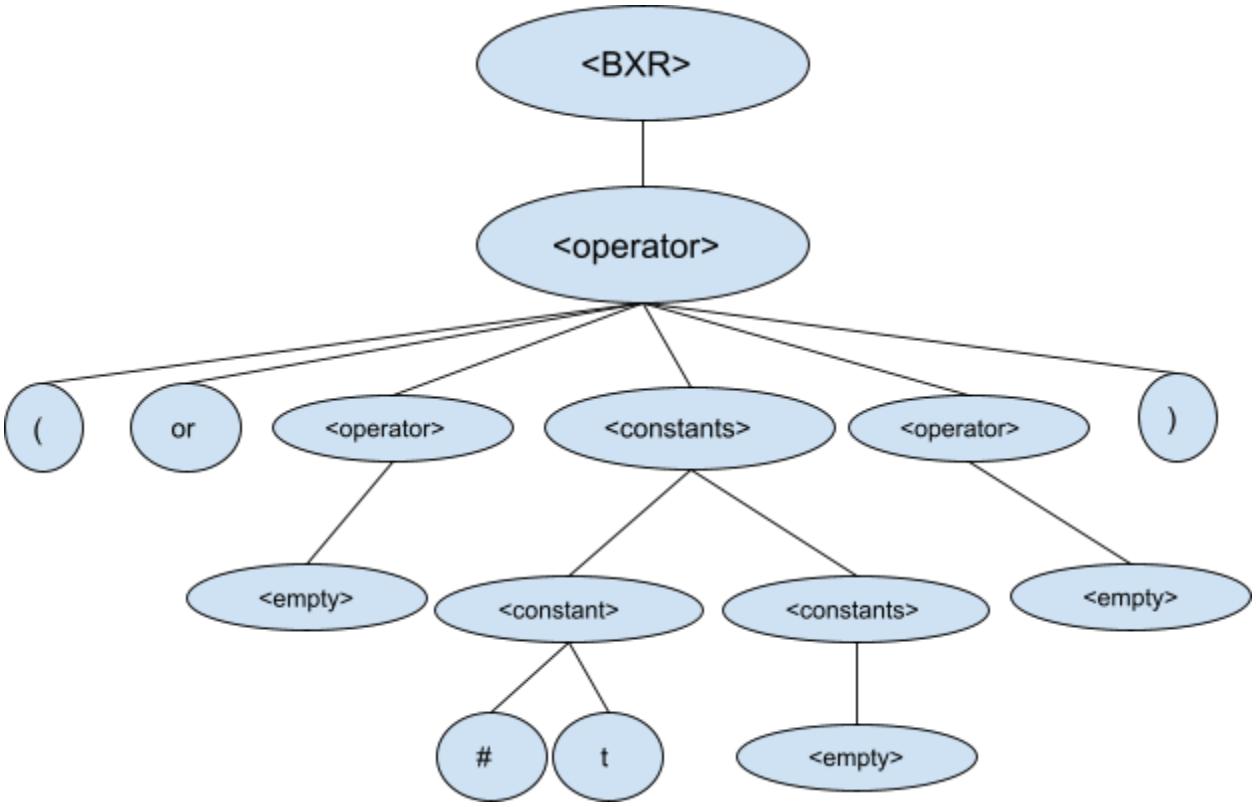
| (not <operator> <constants> <operator>)

<constants> : : = <empty> | <constant> <constants>

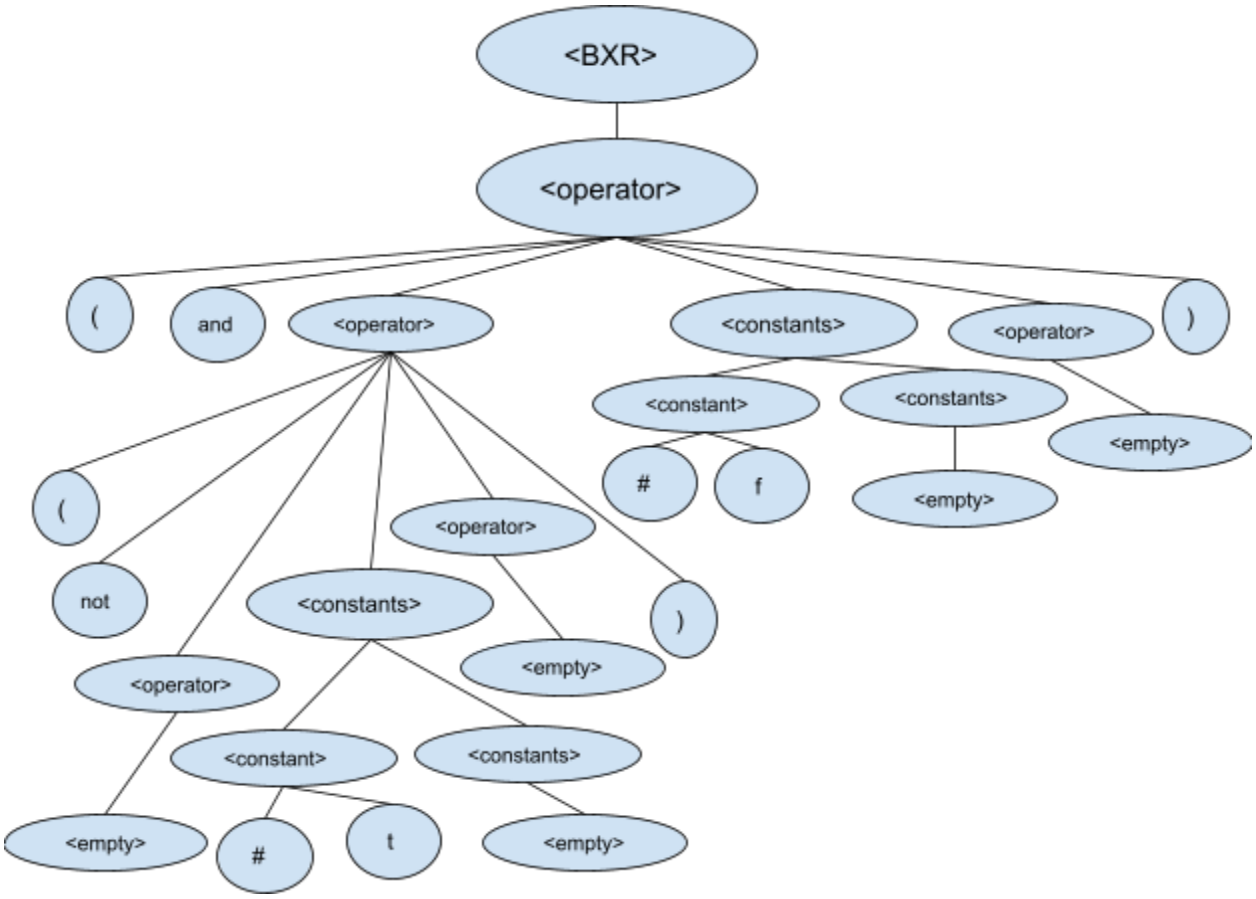
<constant> : : = <empty> | #t | #f

BNF Parse Trees of the BXR language

1. (or #t)



2. (and (not #t) #f)



Problem 5 - CF (Color Fun)

BNF Grammar description of the CF language

<CF> : : = <command>

<command> : : = < empty>

| add (<number> <number> <number> <special-no>) <colors>

| <colors>

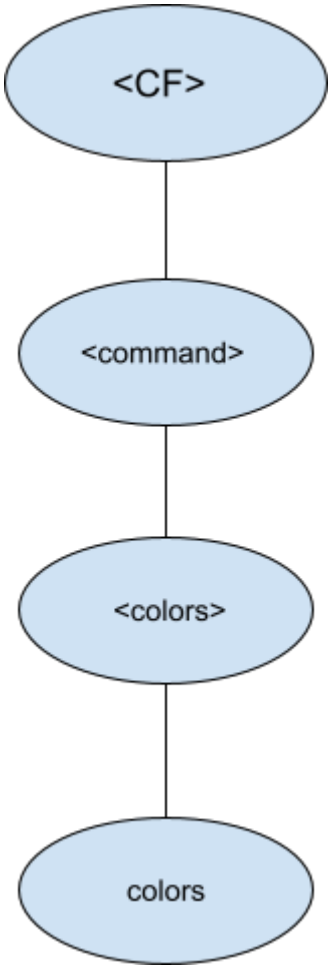
| describe <colors>

| show <colors>

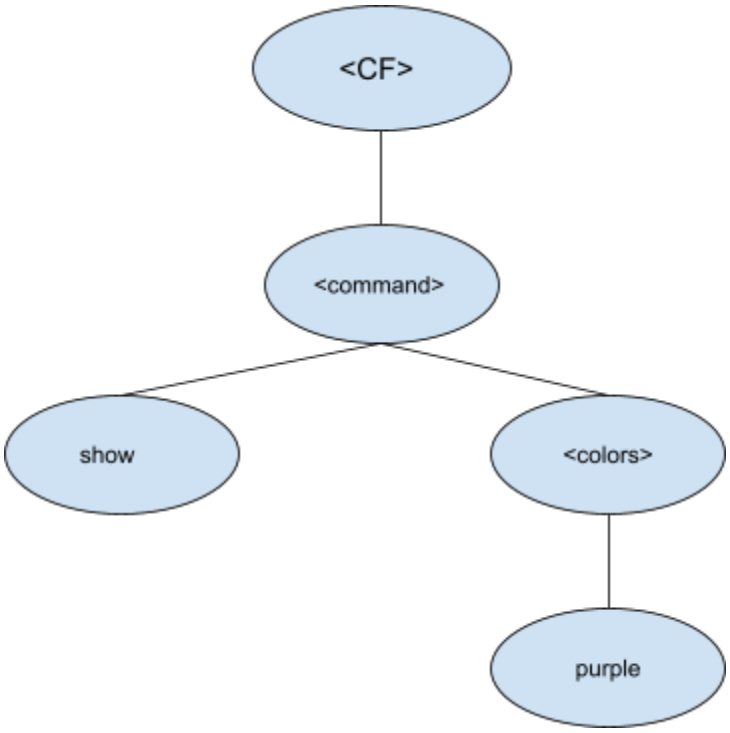
<special-no> : : <empty> | <number>

BNF Parse Trees of the CF language

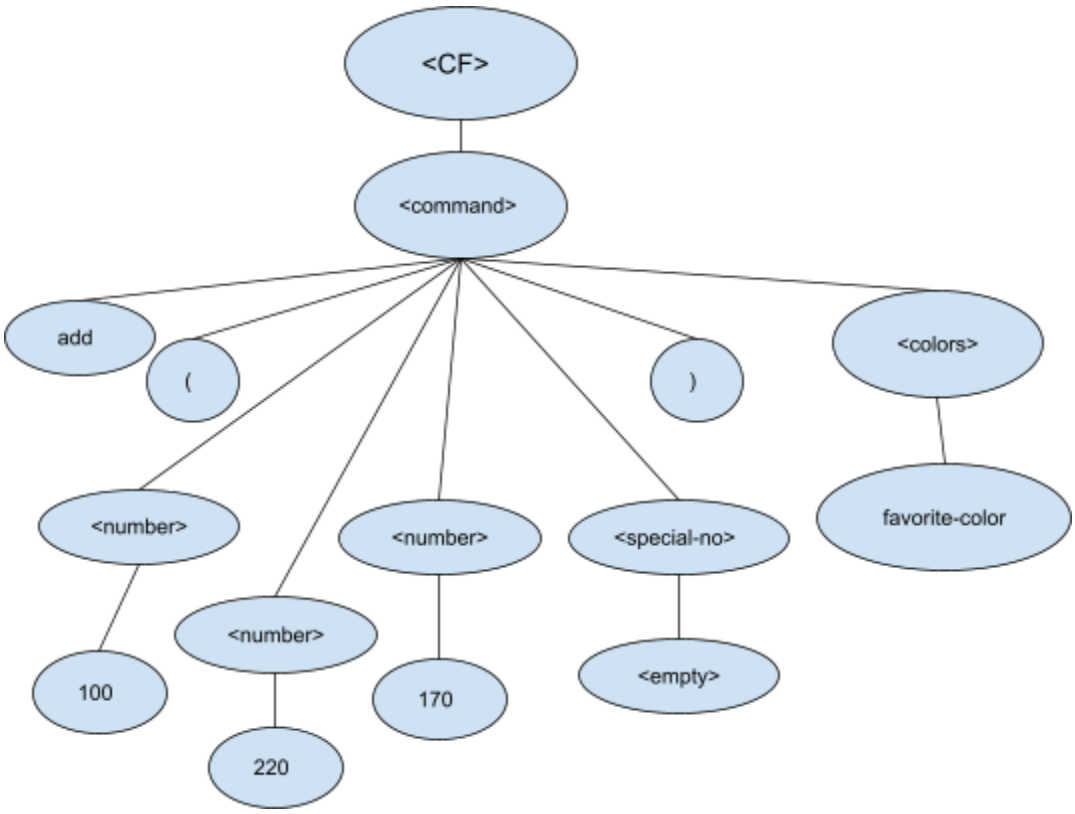
1. colors



2. show purple



3. add (100 220 170) favorite-color



Problem 6 - BNF?

BNF is simply any sentence made up of grammars, verbs, vocabularies and other types of words which plays a vital role in looking at how the function does and how it works in a consecutive and logical way as well as creating the structure of a programming language in a programming world.