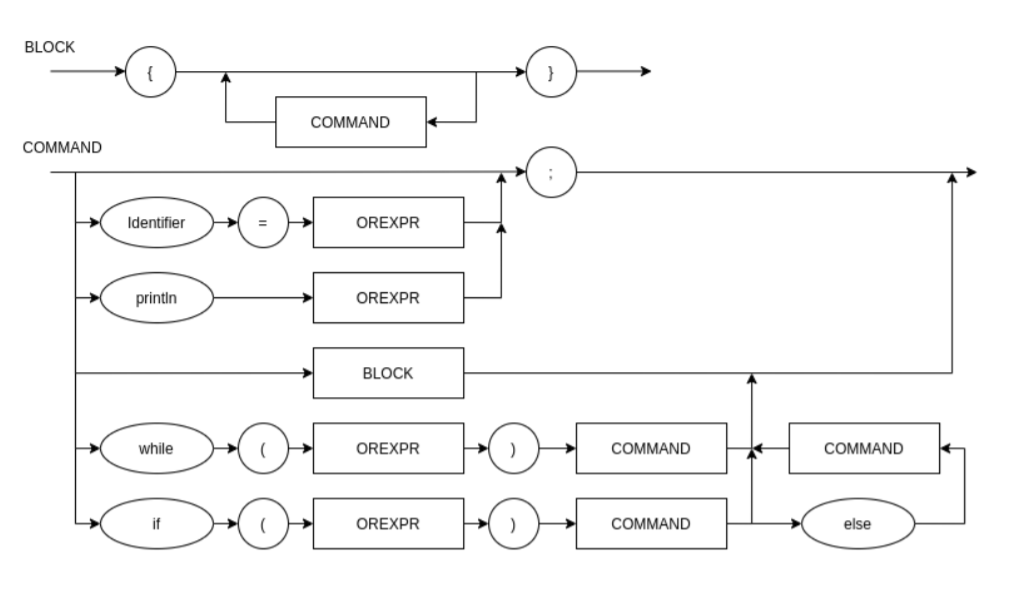
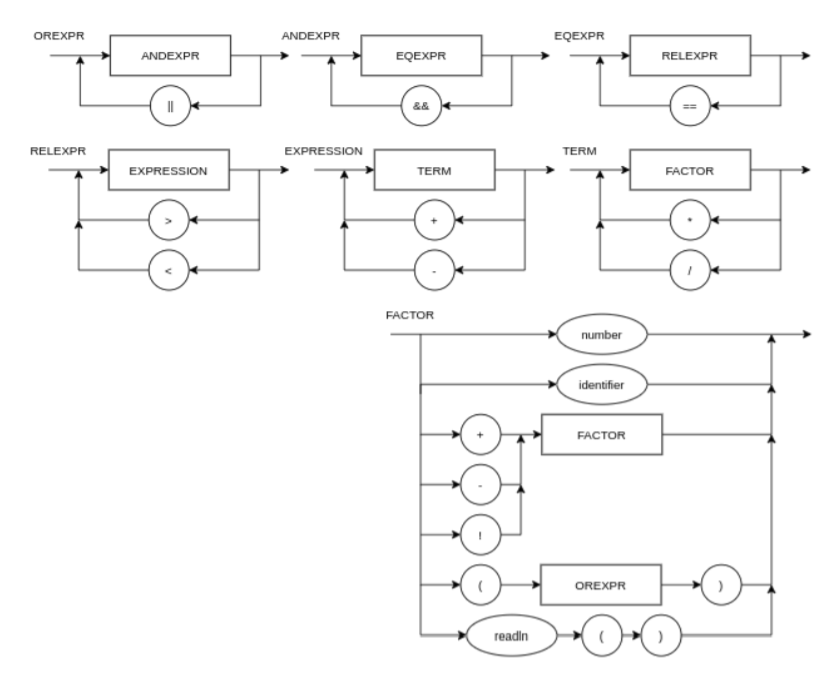
**Roteiro 6 – Lógica da Computação**

**INSPER – 7° Semestre – Matheus Pellizzon**

**Problemas preliminares**

1. Rascunhe as modificações no modelo **EBNF** e no **Diagrama Sintático** baseado nos novos elementos.





1. Liste e explique como serão os novos elementos da AST (valor, quantos filhos, qual a ação, etc).

LogicalOp: classe que extende node. Semelhante ao BinOp, porém para operações lógicas: >, <, ==, || (or), && (and).

- value: a operação

- child: 2 filhos 🡺 child1 <operação> child2

UnOp: foi atualizado para aceitar “!” (not). Nada de seu funcionamento mudou.

Read: classe que extende node. Não precisa de valor nem de filhos. Existe para receber um input do usuário, via terminal. Exemplo;

- x = readln();

def Evaluate(self):

        return int(input("Insert a number: "))

While: o valor que o nó recebe é irrelevante. Possui 2 filhos, um que funciona como a condição para o while, e outro que é o bloco ou comando para ser executado no loop.

def Evaluate(self):

        # conditional child

        while self.children[0].Evaluate():

            # block or command child

            self.children[1].Evaluate()

If: valor que recebe é irrelevante. Pode ter 2 ou 3 filhos (caso exista “else”)

def Evaluate(self):

        # conditional child

        if len(self.children) == 2:

            if self.children[0].Evaluate():

                # block or command child 1

                self.children[1].Evaluate()

            else:

                pass

        else:

            if self.children[0].Evaluate():

                # block or command child 1

                self.children[1].Evaluate()

            else:

                # block or command child 2

                self.children[2].Evaluate()

**Base de Testes:**

Proponha um **programa** de testes, com **TODOS** os seguintes elementos:

• While com and/or/not

• If/Else com and/or/not

• Leitura do terminal

Teste utilizado:

{

    i = 0;

z = 3;

    while (i < 4 && z == 3)

    {

        x = readln();

        if (x > 2 && i > 1)

        {

            println(x);

        }

        else if (!i)

        {

            println(i);

            x = x + --!i;

            println(x);

        }

        else if (!!x || !x)

        {

            println(!!x);

            x = x + x + x;

            println(x);

        }

        i = i + 1;

        println(i);

    }

}