**Вариант 13**

1. Определить в БНФ понятие «натуральное\_число»

<натуральное\_число> ::= <цифра> | <цифра> <натуральное\_число>

<цифра> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

1. Определить в БНФ описание оператора while языка С++

<цикл\_while> ::= "while" "(" <условие> ")" <оператор>

<**условие**> ::= <логическое\_выражение>

<**оператор**> ::= "{" <блок\_операторов> "}"| <простой\_оператор>

<**блок\_операторов**> ::= <оператор> | <оператор> <блок\_операторов>

<**простой\_оператор**> ::= <выражение> ";"| <цикл\_while>| <условный\_оператор>| <оператор\_выбора>| <оператор\_прерывания>| <оператор\_продолжения>

<**логическое\_выражение**> ::= <выражение\_1> | <выражение\_1> <логический\_оператор> <логическое\_выражение>

<**выражение**> ::= <переменная> <оператор\_присваивания> <выражение\_1>| <выражение\_1>

<**выражение\_1**> ::= <выражение\_2>| <выражение\_2><арифметический\_оператор> <выражение\_1>

<**выражение\_2**> ::= <значение>| <переменная>

<**арифметический\_оператор**> ::= "+" | "-" | "\*" | "/"

<**логический\_оператор**> ::= "&&" | "||" | "!"

<**оператор\_присваивания**> ::= "="

<**условный\_оператор**> ::= "if" "(" <логическое\_выражение> ")" <оператор> | "if" "(" <логическое\_выражение> ")" <оператор> "else" <оператор>

<**оператор\_выбора**> ::= "switch" "(" <выражение> ")" "{"<список\_переключателей> "}"

<**список\_переключателей**> ::= <переключатель> | <переключатель> <список\_переключателей>

<**переключатель**> ::= "case" <значение> ":" <блок\_операторов> | "default" ":" <блок\_операторов>

<**оператор\_прерывания**> ::= "break" ";"

<**оператор\_продолжения**> ::= "continue" ";"

<**переменная**> ::= <идентификатор>

<**значение**> ::= <целое\_число> | <дробное\_число> | <логическое\_значение>

<**идентификатор**> ::= <буква> <идентификатор\_продолжение>

<**идентификатор\_продолжение**> ::= <буква> | <цифра> | ""

| <идентификатор\_продолжение> <буква> | <идентификатор\_продолжение> <цифра> | <идентификатор\_продолжение> ""

<**буква**> ::= "a" | "b" | "c" | ... | "z" | "A" | "B" | "C" | ... | "Z"

<**цифра**> ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

G = <T, N, P, S>

N = {S, T}

T =F | TF

F = {'0-9'}

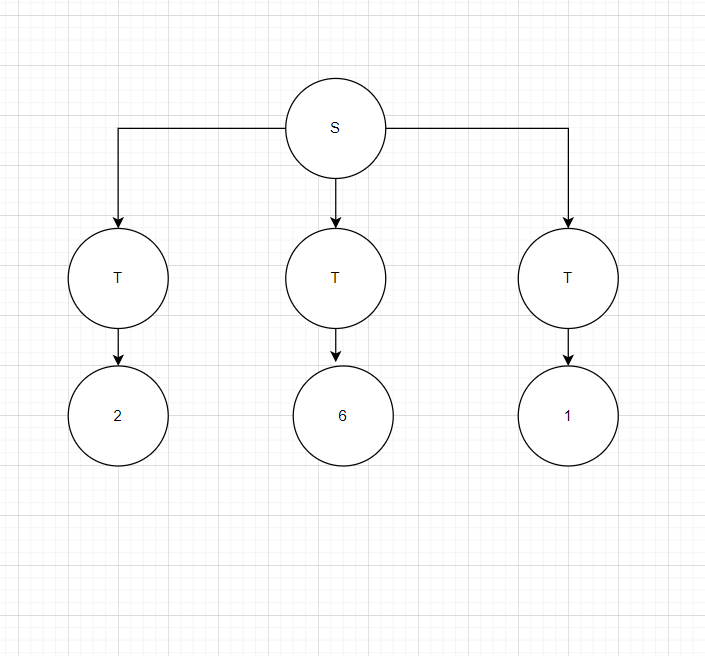
P = {T}

L(G) = {Sx | x ∈ T\*}

пример вывода правильной цепочки языка:

S -> T -> TF -> TFF -> TTT -> 2TT -> 26T ->261

S -> T-> TF-> TF2 -> T62 -> 162



Данная грамматика является контекстно-свободной, так как все её правила имеют вид T -> α, где T - нетерминальный символ, а α - цепочка символов, не зависящая от контекста или окружения, в котором T находится. В данной грамматике все нетерминальные символы S, D, T слева от стрелки в правилах, и они заменяются на соответствующие цепочки символов. Таким образом, данная грамматика удовлетворяет определению контекстно-свободной грамматики и не превышает её ограничения.