start → expr ; expr → expr + expr

```
bool A() {
    for each production A -> X1 X2 ... Xk {
        matched_all = true;
        for i in 1 to k {
            if Xi is a nonterminal {
                succeeded = Xi();
                if !succeeded {
                    matched_all = false;
                    break;
                }
            } else if Xi == current input token {
                // advance
            } else {
                matched_all = false;
                break;
            }
        }
        if matched_all {
            return true;
        }
    }
    return false;
}
```

input: 1 + 2 + a

↑

A → ε

start()

expr()

expr()

A → B ....

B → C ...

C → D ...

D → A

A()

B()

C()

D()

A()

A → Aα

A → β

α at least 1 symbol

the first symbol of β ≠ A

A → βA'

A' → αA' | ε

$N_{rx} \rightarrow \underbrace{N_{rx}}_{A} \underbrace{N_r}_{\alpha} \mid \varepsilon$

$N_{rx} \rightarrow \underbrace{\varepsilon}_{\beta}$

~~$N_{rx} \rightarrow \varepsilon N_{rx}'$~~

$N_{rx}' \rightarrow N_r N_{rx}' \mid \varepsilon$

$$A \to A\alpha_1 \mid A\alpha_2 \mid \dots \mid A\alpha_k \mid \beta_1 \mid \beta_2 \mid \dots \mid \beta_\ell$$

$$A \to \beta_1 A' \mid \beta_2 A' \mid \dots \beta_\ell A'$$

$$A' \to \alpha_1 A' \mid \alpha_2 A' \mid \dots \alpha_k A' \mid \varepsilon$$

$$\underbrace{expr}_{A} \to expr \underbrace{+ expr}_{\alpha_1} \mid expr \underbrace{- expr}_{\alpha_2} \mid \underbrace{factor}_{\beta_1}$$

$$factor \to (expr) \mid ID \mid NUM$$

$$expr \to factor\ expr'$$

$$expr' \to \underbrace{+ expr}_{\alpha_1}\ expr' \mid \underbrace{- expr}_{\alpha_2}\ expr' \mid \varepsilon$$

$$factor \to (expr) \mid ID \mid NUM$$

start
expr → factor expr'

expr' → + expr expr' | - expr expr' | ε
(+ expr: α_1; - expr: α_2)

factor → (expr) | ID | NUM

input

1 + (2 - a) + b

↑

expr()

factor()

try factor → (expr) $x_1\ x_2\ x_3$

try factor → ID x_1'

try factor → NUM

return true

expr'() $x_1\ x_2\ x_3$

try expr' → + expr expr'

expr()

factor()

expr()

factor()