

C— 语言的 LL(1) 文

文法的开始符号：S

空字符串：\$

终结符：

- 关键字: 包括 while break continue for if else float int void return main 算符包括+ - * / = >= == <=
- 界符包括 () { } ;
- 标识符 IDN , 常数的定义与 C 语言相同 : INT 、 FLOAT

具体文法如下：

```
S -> funcs // $ 为 epsilon, S为文法的开始符号
funcs -> func funcs
funcs -> $

// func 可以推出函数和define_stmts
func -> void IDN ( arg ) func_body // IDN identifier 标识符
func -> IDN init ;
func -> type IDN funordefine
funordefine -> ( arg ) func_body
funordefine -> init ;
// type 为普通数据类型 ---- int float
type -> int
type -> float
arg -> type IDN //因为没有', '界符, 所以只能有一个或者零个参数
arg -> $

func_body -> ;
func_body -> block; // block 函数体, 内部由各种语句构成
block -> { statements }
statements -> statement statements
statements -> $ // 函数体可为空
// 语句分为三种, 赋值、定义变量语句, 控制语句, 调用函数语句 statement ->
define_stmts
statement -> init_exp ;
statement -> control_stmts
fun_value_exp -> $
fun_value_exp -> init
fun_value_exp -> ( isnull_expr )
init -> = expression
init -> $

// 控制语句分为: 跳转、循环、分支
control_stmts -> jump_stmt // 跳转 jump_stmt
```

```

control_stmts -> iteration_stmt // 循环(for、while)
control_stmts -> if_control_stmts
// 分支支(if)
if_control_stmts -> if ( expression ) block_stmt else_result
// 跳转(continue、break、return)
jump_stmt -> continue ;
jump_stmt -> break ;
jump_stmt -> return isnull_expr ;
// 循环(for、while)
iteration_stmt -> while ( expression ) block_stm
iteration_stmt -> for ( null_init_exp ; isnull_expr ; isnull_expr )
block_stmt
null_init_exp -> init_exp
null_init_exp -> $
else_result -> else elseif_stmt
elseif_stmt -> if_control_stmts
elseif_stmt -> block_stmt
else_result -> $
block_stmt -> block
block_stmt -> statement
isnull_expr -> expression
isnull_expr -> $

block_stm -> block // 可以没有花括号, 用用block_stm表示代码块 block_stm ->
statements // 没有外面面的花括号

// expression 逻辑表达式:1 value (>=、<=、==、=) value 2 value expression ->
value operation
expression -> value exp
exp -> $
exp -> op value

op -> >=
op -> <=
op -> ==
op -> =

// value 算数表达式 多个item进行行行加减乘除组合 value -> item value1
value1 -> + item value1
value1 -> - item value1
value1 -> / item value1
value1 -> * item value1
value1 -> $

// item有三种情况:(value)、调用用函数的返回值 IDN call_func、常数 item -> ( value
)
item -> IDN fun_value // 函数返回值
// 常数, 分为INT 为整数(1、4、12), FLOAT为浮点数(10.2e)
item -> ( value )

```

```
item -> INT
item -> FLOAT
fun_value -> $
fun_value -> ( isnull_expr )
```

func 可以推出函数和定义、初始化语句。

block 函数体，内部由各种语句构成语句分为三种 define_stmts 赋值、定义变量语句 int a = 1;g=10;
func_stmts 调用用函数语句句control_stmts 控制语句句

控制语句分为:(可以没有花括号，用用block_stm表示代码块) 跳转(continue、break、return) 循环
(for、while) 分支支(if)

isnull_expr 可以推出逻辑表达式和空 expression 逻辑表达式:1 value (>=、<=、==、=) value 2
value value 算数表达式 有item进行行行加减乘除组合 item :(value)、调用用函数的返回值、常数，
分为INT为整数(1、4、12)，FLOAT为浮点数(10.2e)

```
// 测试用例:
void main(){
    int i=2;
    int j=4;
    int result=0;
    while(i <=j){
        result=result+j;
        continue;
    }
    for(int i = 0;i<=10;i=i+1){
        j = i+1;
    }

    printf();
    return result;
}
```