# 文法G

|  |  |  |
| --- | --- | --- |
| Id | 推导式 | 说明 |
| 1 | P ->PC DL | 程序推出前置代码和声明列表 |
| 2 | PC->HL NL | 前置代码推出头文件列表或者名字空间列表 |
| 3 | HL->HL #include<Id>|  HL #include”Id”|  #include<Id>|  #include”Id” | 头文件列表推出头文件列表和头文件或头文件 |
| 4 | NL->NL using namespace Id|  using namespace Id | 名字空间列表推出名字空间列表和名字空间或者名字空间 |
| 5 | DL->DL D|D | 声明列表推出声明列表和声明或者声明 |
| 6 | D ->Dv|Df | 声明推出变量声明或者程序声明 |
| 7 | Dv->T Id | 变量声明推出类型符号和标识符 |
| 8 | T -> int|float|double|char|void | 类型符号推出常用的类型符 |
| 9 | Df->T Id(Pm) CS | 函数声明推出类型、函数名标识符、参数列表和代码块 |
| 10 | Pm->PL|void | 函数参数推出参数列表或者void |
| 11 | PL->PL,p|p | 参数列表推出参数列表，参数或者参数 |
| 12 | p->T Id | 参数推出类型符号和标识符 |
| 13 | CS->{Dl SL} | 函数体推出局部变量声明和语句列表 |
| 14 | Dl->Dl Dv|E | 局部变量声明推出局部变量声明和变量声明或者表达式 |
| 15 | SL->SL S|E | 语句列表推出语句列表和语句或者表达式 |
| 16 | S->ES|CS|SS|IS|RS | 语句推出表达式语句|代码块|选择语句|迭代语句|返回语句 |
| 17 | ES->E; | 表达式语句推出表达式和分号 |
| 18 | SS->if (E) S|if (E) S else S | 选择语句推出if…或者if…else… |
| 19 | IS->while (E) S | 迭代语句推出while… |
| 20 | RS ->return;|return E | 返回语句推出return空或者表达式 |
| 21 | E->Id=E|Es | 表达式推出标识符赋值或者简单表达式 |
| 22 | Es->Ea relop Ea|Ea | 简单表达式推出两个表达式的r运算或者表达式 |
| 23 | relop-><= |<|>|>=|==|!=|&&| || | 推出比较运算符和逻辑算术运算符 |
| 24 | Ea->Ea addop Eb|Eb | 表达式a推出表达式ab之间的a运算或者表达式b |
| 25 | addop->+|- | a操作推出加减 |
| 26 | Eb->Eb mulop Ec | Ec | 表达式b推出表达式bc之间的m运算或者表达式c |
| 27 | mulop->\* | / | m操作推出乘除 |
| 28 | Ec->(E) | Id | NUM | 表达式c推出带括号表达式或者变量或常量 |

# 文法

|  |  |  |
| --- | --- | --- |
| Id | 推导式 | 说明 |
| 1 | P ->PC DL | 程序推出前置代码和声明列表 |
| 2 | PC->HL NL | 前置代码推出头文件列表或者名字空间列表 |
| 3 | HL->#include<Id> H1 | 头文件列表推出头文件和H1 |
| 4 | HL->#include”Id”H1 |  |
| 5 | H1->HL | 提取公共左因子 |
| 6 | H1->ε |  |
| 7 | NL->using namespace Id N1 | 名字空间列表推出名字空间和N1 |
| 8 | N1->NL | 提取公共左因子 |
| 9 | N1->ε |  |
| 10 | DL->D D0 | 声明列表推出声明和D1 |
| 11 | D0->DL | 提取公共左因子 |
| 12 | D0->ε |  |
| 13 | D ->T Id Dt | 声明推出变量声明或者程序声明 |
| 14 | Dt->Df |  |
| 15 | Dt->; |  |
| 16 | T -> int | 类型符号推出常用的类型符 |
| 17 | T -> float |  |
| 18 | T -> double |  |
| 19 | T -> char |  |
| 20 | T -> void |  |
| 21 | Df->(Pm) CS | 函数声明推出类型、函数名标识符、参数列表和代码块 |
| 22 | Pm->PL | 函数参数推出参数列表 |
| 72 | Pm->ε |  |
| 23 | PL->p P1 | 参数列表推出参数和P1 |
| 24 | P1->,PL | 提取公共左因子 |
| 25 | P1->ε |  |
| 26 | p->T Id | 参数推出类型符号和标识符 |
| 27 | CS->{Dl SL} | 函数体推出局部变量声明和语句列表 |
| 28 | Dl->T Dl | 局部变量声明推出变量声明和局部变量声明或者表达式 |
| 29 | Dl->E |  |
| 30 | SL->S Sp | 语句列表推出语句和语句列表或者表达式 |
| 31 | Sp->SL |  |
| 32 | Sp->ε |  |
| 33 | S->ES | 语句推出表达式语句|代码块|选择语句|迭代语句|返回语句 |
| 34 | S->CS |  |
| 35 | S->SS |  |
| 36 | S->IS |  |
| 37 | S->RS |  |
| 38 | ES->E; | 表达式语句推出表达式和分号 |
| 39 | SS->if (E) S S1 | 选择语句推出if…S…S1 |
| 40 | S1->else S | 提取公共左因子 |
| 41 | S1->ε |  |
| 42 | IS->while (E) S | 迭代语句推出while… |
| 43 | RS ->return R1 | 返回语句推出return和R1 |
| 44 | R1->; | 提取公共左因子 |
| 45 | R1->E |  |
| 46 | E->Id=E; | 表达式推出标识符赋值或者简单表达式 |
| 47 | E->Es |  |
| 48 | Es->Ea Es1 | 简单表达式推出表达式a和Es1达式 |
| 49 | Es1->relop Ea | 提取公共左因子 |
| 50 | Es1->ε |  |
| 51 | relop-><= | 推出比较运算符和逻辑算术运算符 |
| 52 | relop->< |  |
| 53 | relop->> |  |
| 54 | relop->>= |  |
| 55 | relop->== |  |
| 56 | relop->!= |  |
| 57 | relop->&& |  |
| 58 | relop->|| |  |
| 59 | Ea-> Eb Ea1 | 表达式a推出表达式b和Ea1 |
| 60 | Ea1->addop Eb Ea1 | 消除左递归 |
| 61 | Ea1->ε |  |
| 62 | addop->+ | a操作推出加减 |
| 63 | addop->- |  |
| 64 | Eb->Ec Eb1 | 表达式b推出表达式c和Eb1 |
| 65 | Eb1->mulop Ec Eb1 | 消除左递归 |
| 66 | Eb1->ε |  |
| 67 | mulop->\* | m操作推出乘除 |
| 68 | mulop->/ |  |
| 69 | Ec->(E) | 表达式c推出带括号表达式或者变量或常量 |
| 70 | Ec->Id |  |
| 71 | Ec->NUM |  |

# First

|  |  |
| --- | --- |
|  | **First** |
| P | #include“,#include< |
| PC | #include“,#include< |
| DL | int,float,double,char,void |
| HL | #include“,#include< |
| NL | using namespace |
| H1 | #include“,#include<,ε |
| N1 | using namespace,ε |
| D | int,float,double,char,void |
| Dt | (,; |
| D0 | int,float,double,char,void,ε |
| Df | ( |
| T | int,float,double,char,void |
| Pm | int,float,double,char,void |
| CS | { |
| PL | int,float,double,char,void |
| p | int,float,double,char,void |
| P1 | ,|ε |
| Dl | int,float,double,char,void,Id,(,NUM |
| SL | (,Id,NUM,{,if,while,return |
| E | (,Id,NUM |
| S | (,Id,NUM,{,if,while,return |
| Sp | (,Id,NUM,{,if,while,return,ε |
| ES | I(,Id,NUM |
| SS | if |
| IS | while |
| RS | return |
| S1 | else,ε |
| R1 | ;, (,Id,NUM |
| Es | (,Id,NUM |
| Es1 | <= ,<,>,>=,==,!=,&&,||,ε |
| relop | <= ,<,>,>=,==,!=,&&,|| |
| Ea | (,Id,NUM |
| Ea1 | +,-,ε |
| addop | +,- |
| Eb | (,Id,NUM |
| Eb1 | \*,/,ε |
| mulop | \*,/ |
| Ec | (,Id,NUM |

# Follow

|  |  |
| --- | --- |
|  | **Follow** |
| H1 | using namespace |
| N1 | int,float,double,char,void |
| D0 | # |
| P1 | ) |
| S1 | (,Id,NUM,{,if,while,return,else |
| Sp | } |
| Es1 | (,Id,NUM,{,if,while,return,else,},),; |
| Ea1 | (,Id,NUM,{,if,while,return,else,},),;,<= ,<,>,>=,==,!=,&&,|| |
| Eb1 | (,Id,NUM,{,if,while,return,else,},),;,<= ,<,>,>=,==,!=,&&,||,+,- |