C1语言定义

1. 块

C1语言的程序由一个块构成。块由“{”开始，由“}”结束，其中包含若干语句。

1. 数据类型

C1语言只包含一种数据类型——整型，即32位有符号整数，取值范围是，类型标识符为“int”。

1. 变量

C1语言只包含整型变量，必须先声明后使用，同名变量不可重复声明。未初始化变量默认值为0。

1. 常量

C1语言只包含整型字面常量，且为十进制整数。

1. 表达式

C1语言的表达式包括数值运算表达式、数值比较表达式和逻辑运算表达式。支持配对嵌套的圆括号“( )”用于控制运算优先级。

由于C1语言的数值类型只有整型，而整型对加法、减法和乘法运算封闭，故C1语言支持的二元数值运算为加法、减法和乘法。数值运算表达式的结果仍为整型。常量、变量均可视作数值运算表达式。

数值比较表达式的结果为真值，不可用于数值运算或进一步比较，只能用于逻辑运算。

C1语言虽然没有布尔型变量，但是支持对整型的逻辑判断，零值判为假，非零值判为真。变量、常量和数值运算表达式之间还可进行或、且、非三种逻辑运算。逻辑运算表达式的结果为真值，不可用于数值运算或比较，只能用于进一步的逻辑运算。

1. 语句

C1语言支持顺序、分支和循环三种控制流结构。包括赋值语句、输入语句和输出语句。输出语句只可输出整型。

1. C1语言的词法

全书使用如下正则定义：

*letter* = a | b | … | z | A | B | … |Z

*digit* = 0 | 1 | … | 9

*underscore* = \_

* 1. 关键字

int if else while input output

保留字，大小写敏感。

* 1. 标识符

*id* = ( *letter* | *underscore* ) ( *letter* | *digit* | *underscore* )\*

* 1. 常量

*const* = *digit digit*\*

* 1. 运算符和分隔符

+ - \* < <= > >= == !=

&& || ! ; ( ) { }

* 1. 空白符

空格（blank）、换行（newline）和制表符（tab）

*whitespace* = ( **blank** | **tab** | **newline** )+

* 1. 注释  
     C1语言支持单行注释，以“//”开头延伸至行尾。

1. C1语言的语法

为书写方便，使用如下缩写：

stmt – statement

decl – declaration

expr – expression

var – variable

const – constant

bi – binary

oper – operator

如下文法是二义的，需结合算符的优先级和结合性规则，以及if – else语句的就进匹配原则来理解。

*program* → *block*

*block* → { *stmts* }

*stmts* → *decl* *stmts*

| *stmt stmts*

| ε

*decl* → int *id* ;

*stmt* → *id* = *expr* ;

| if ( *expr* ) *stmt*

| if ( *expr* ) *stmt* else *stmt*

| while ( *expr* ) *stmt*

| input ( *id* ) ;

| output ( *expr* ) ;

*| block*

*expr* → *id*

| *const*

| ( *expr* )

| *unary\_oper expr*

| *expr* *val\_oper* *expr*

*unary\_ oper* →+ | - | !

*bi\_oper* → + | - | \* | < | <= | > | >= | == | != | || | &&

Token类设计

所有记号（token）都需要为外界提供获得行号、列号、单词（lexeme）和附加属性的方法。附加属性对不同的记号可能不同，对常量而言是整型，对标识符而言是数据类型，等等；如果有更多需要，还可以是自定义类。为了统一约定这些规则，使用