

Oberon-0

Oberon-0语言特点

保留字与关键字

- 保留字(Reserved word)是在编程语言中有特殊意义的词不能有其它用途。这是一个语法层次的概念，保留字有语法上的意义，例如在Pascal中 `if` 用于组成条件语句。很多保留字都满足特定编程语言的标志符的规定，但是如果被定义为保留字，那么就不能再被用来当作标志符，否则会引起语法错误。
- 关键字(Keyword)同样是在编程语言中有特殊意义的词，但是与保留字不同，关键字只有在特殊的上下文中才有特殊的意义，它是一个语义概念。例如在Fortran语言里 `if (if > 0) then a = 1 end if` 是一条合法的语句，第一个 `if` 会被解析为条件语句的起始，第二个则是一个标志符。

Oberon-0的表达式与C/C++等语言的区别

Oberon-0文法二义性

文档中Oberon-0的BNF定义是无二义的。

通常编程语言会遇到的二义性问题有以下两种

- 操作符的优先级与结合律。有些文法的BNF不会规定优先级和结合律，但是Oberon-0的BNF通过 `term`, `factor` 两个非终结符解决了优先级的问题。每个终结符的产生式都通过一定规则规定了结合律为左结合，因此在这方面Oberon-0是无二义的。
- 悬挂 `else`。像C/C++这样的语言在遇到下面这段代码的时候会遇到 `else` 与哪一个 `if` 匹配的问题。但是Oberon-0不会有这种问题，因为它有 `END` 关键字。

```
if boolean
    if boolean
        statement
else
    statement
```

如果 `else` 与第一个 `if` 匹配，那么只能写成下面这样：

```
IF boolean THEN
    IF boolean THEN
        statement
    END
ELSE
    statement
END
```

如果 `else` 与第二个 `if` 匹配，那么只能写成下面这样：

```
IF boolean THEN
  IF boolean THEN
    statement
  ELSE
    statement
  END
END
```

所以 `END` 起到了匹配的作用。类似地，C/C++等语言也可以通过大括号 `{}` 来明确 `else` 的匹配问题，当然这些语言已经规定了最近匹配原则。

综合上述讨论，Oberon-0在文法的定义上解决了二义性问题。

心得

在二义性讨论的时候认识到常见的二义性问题可能在语言设计的时候就避免了，例如Oberon-0的else问题。