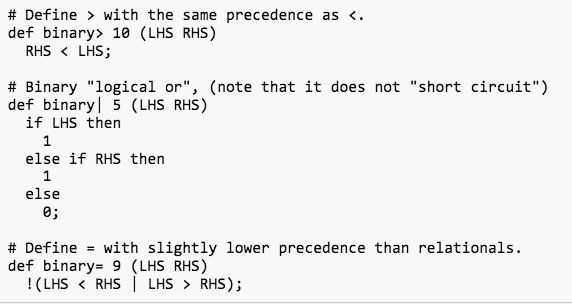
GitHub网址：

https://github.com/Interpretion/Interpreter.git

用户定义的运算符

主要就是：添加自定义的运算符：重定义存在的运算符、以编程的方式改变语法规则、引入新的运算符、更改优先级。

教程中举了例子：增加一元运算符以及二元运算符。因为Kaleidoscope没有一元运算符。以及修改<,>的优先级等。

implement standard runtime library in the language itself：①实现对自定义的二元运算符的支持；②添加一元运算符。

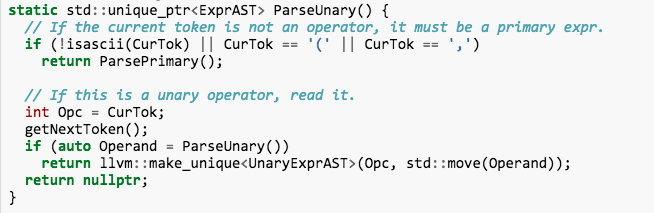
实现对自定义的二元运算符的支持：首先增加对一元/二元运算符关键词的支持。



对关键字的词法分析器的支持

为了将自定义的运算符表示为原型，扩展AST节点。跟踪它是否是运算符，如果是，优先级是什么？解析运算符，为已有的二元运算符添加一些默认情况。最后，before codegening a function，如果是用户自定义的运算符，就需要在优先级表中注册，这将允许二元运算符解析已有的逻辑，所以我们需要做的就是扩展语法。

自定义的一元运算符：创建一个AST节点，添加解析逻辑（一元运算符不需要优先级信息）；



为原型添加解析器支持，以解析一元运算符原型。最后，添加对一元运算符的codegen支持。