

编译原理 Lab6 实验报告

姓名：熊丘桓

学号：201250127

邮箱：eaglebear@smail.nju.edu.cn

1. 实现功能

本次实验完成了以下功能：

1. 全局变量声明和使用
2. 条件语句

2. 实验设计

本次实验重新设计了 LLVMVisitor 类，其中全局变量相关内容主要是对 visitVarDecl 方法的修改：

```

1  @Override
2  public LLVMValueRef visitVarDecl(SysYParseVarDeclContext ctx) {
3      String typeName = ctx.bType().getText();
4
5      for (SysYParseVarDefContext varDefContext : ctx.varDef()) {
6          LLVMTypeRef varType = getTypeRef(typeName);
7          String varName = varDefContext.IDENT().getText();
8          int elementCount = 0;
9
10         for (SysYParseConstExpContext constExpContext : varDefContext.constExp()) {
11             elementCount = Integer.parseInt(toDecimalInteger(constExpContext.getText()));
12             varType = LLVMArrayType(varType, elementCount);
13         }
14
15         LLVMValueRef varPointer = null;
16         if (currentScope == globalScope) {
17             varPointer = LLVMAddGlobal(module, varType, "pointer_" + varName);
18             if (elementCount == 0) {
19                 LLVMSetInitializer(varPointer, zero);
20             } else {
21                 PointerPointer<Pointer> pointerPointer = new PointerPointer<>
22                 (elementCount);
23                 for (int i = 0; i < elementCount; ++i) {
24                     pointerPointer.put(i, zero);
25                 }
26                 LLVMValueRef initArray = LLVMConstArray(varType, pointerPointer,
27                 elementCount);
28                 LLVMSetInitializer(varPointer, initArray);
29             }
30         } else {
31             varPointer = LLVMBuildAlloca(builder, varType, "pointer_" + varName);
32         }
33
34         if (varDefContext.ASSIGN() != null) {
35             SysYParseExpContext expContext = varDefContext.initVal().exp();
36             if (expContext != null) {
37                 LLVMValueRef initVal = visit(expContext);
38                 if (currentScope == globalScope) {

```

```

37         LLVMSetInitializer(varPointer, initVal);
38     } else {
39         LLVMBuildStore(builder, initVal, varPointer);
40     }
41 } else {
42     int initValCount = varDefContext.initVal().initVal().size();
43     if (currentScope == globalScope) {
44         PointerPointer<Pointer> pointerPointer = new PointerPointer<>
(elementCount);
45         for (int i = 0; i < elementCount; ++i) {
46             if (i < initValCount) {
47                 pointerPointer.put(i,
this.visit(varDefContext.initVal().initVal(i).exp()));
48             } else {
49                 pointerPointer.put(i, zero);
50             }
51         }
52         LLVMValueRef initArray = LLVMConstArray(varType, pointerPointer,
elementCount);
53         LLVMSetInitializer(varPointer, initArray);
54     } else {
55         LLVMValueRef[] initArray = new LLVMValueRef[elementCount];
56         for (int i = 0; i < elementCount; ++i) {
57             if (i < initValCount) {
58                 initArray[i] =
this.visit(varDefContext.initVal().initVal(i).exp());
59             } else {
60                 initArray[i] = LLVMConstInt(i32Type, 0, 0);
61             }
62         }
63         buildGEP(elementCount, varPointer, initArray);
64     }
65 }
66 }
67
68     currentScope.define(varName, varPointer);
69 }
70
71     return null;
72 }

```

3. 实验困难

笔者实验过程中仍有三个困难的测试用例无法解决，报错信息如下：

```

1     hardtest1.sy: 0 (Aborted (core dumped) )
2     hardtest2.sy: 0 (Aborted (core dumped) )
3     hardtest3.sy: 0 (Aborted (core dumped) )

```