

May 07, 15 9:11

## ManyReallyLongLines.c

Page 1/2

```

case Var_idK: {
    varFlag = 1;
    recordSym = getSymbol(e->t, e->val.Var_idE.Var_idE);
    if (lValue) {
        if (recordSym->scopeType == scopeGlobal) {
            if ((recordSym->typeSymbol->nodetype==record && subRoutFlag) || (recordSym->typeSymbol->nodetype==array && indexFlag)) {
                insertCode(codeList, "\tmovl %s, %%eax \t#Move the content %s into %%eax\n", e->val.Var_idE.Var_idE, e->val.Var_idE.Var_idE); //We need the content, so we can store the address of the array or record there
            } else {
                insertCode(codeList, "\tlea %s, %%eax \t#Move the address of %s into %%eax\n", e->val.Var_idE.Var_idE, e->val.Var_idE.Var_idE); //We need logical address of local vars
            }
            insertCode(codeList, "\tpushl %%eax\n");
        } else {
            if ((recordSym->typeSymbol->nodetype==record && subRoutFlag) || (recordSym->typeSymbol->nodetype==array && indexFlag)) {
                insertCode(codeList, "\tmovl %d(%%ebp), %%eax \t#Move the content of the parameter %s at offset %d into %%eax\n", recordSym->stackOffset, e->val.Var_idE.Var_idE, recordSym->stackOffset); //We need the content, so we can store the address of the array or record there
                insertCode(codeList, "\tpushl %%eax \t#Push %%eax onto the stack\n");
            } else {
                if (recordSym->nested == nestedFunc) {
                    insertCode(codeList, "\tlea %d(%%ebp), %%eax \t#Move the address of the parameter/local variable %s at offset %d into %%eax\n", recordSym->stackOffset, e->val.Var_idE.Var_idE, recordSym->stackOffset);
                    insertCode(codeList, "\tpushl %%eax \t#Push %%eax onto the stack\n");
                } else if (recordSym->nested < nestedFunc) { //if we have nested functions we need to calculate another offset for the params
                    int diff = nestedFunc-recordSym->nested-1;
                    insertCode(codeList, "\tmovl (%%ebp), %%ebx \t#Finding the static link to the parent function\n");
                    for (int i = 0; i < diff; i++) {
                        insertCode(codeList, "\tmovl (%%ebx), %%ebx \t#Finding the static link to the parent function\n");
                    }
                    insertCode(codeList, "\tlea %d(%%ebx), %%eax \t#Move the address of the parameter/local variable %s at offset %d into %%eax\n", recordSym->stackOffset, e->val.Var_idE.Var_idE, recordSym->stackOffset);
                    insertCode(codeList, "\tpushl %%eax \t#Push %%eax onto the stack\n");
                }
            }
        } else {
            if (recordSym->scopeType == scopeGlobal) {
                insertCode(codeList, "\tmovl %s, %%eax \t#Move the content of %s into %%eax\n", e->val.Var_idE.Var_idE, e->val.Var_idE.Var_idE);
                insertCode(codeList, "\tpushl %%eax \t#Push %%eax onto the stack\n");
            } else if (recordSym->scopeType == scopeFunctionParam || recordSym->scopeType == scopeFunctionLocal) {
                if (recordSym->nested == nestedFunc) {
                    insertCode(codeList, "\tmovl %d(%%ebp), %%eax \t#Move the local vari

```

May 07, 15 9:11

## ManyReallyLongLines.c

Page 2/2

```

able/parameter %s at offset %d into %%eax\n", recordSym->stackOffset, e->val.Var_idE.Var_idE
,recordSym->stackOffset);
        insertCode(codeList, "\tpushl %%eax        \t#Push %%eax onto the stac
k\n");
        } else if (recordSym->nested < nestedFunc) { //if we have ne
sted functions we need to calculate another offset for the params
        int diff = nestedFunc-recordSym->nested-1;
        insertCode(codeList, "\tmovl (%%ebp), %%ebx  \t#Finding the static li
nk to the parent function\n");
        for (int i = 0; i < diff; i++) {
            insertCode(codeList, "\tmovl (%%ebx), %%ebx  \t#Finding the st
atic link to the parent function\n");
        }
        insertCode(codeList, "\tmovl %d(%%ebx), %%eax \t#Move the content o
f the variable/parameter %s at offset %d into %%eax\n", recordSym->stackOffset, e->val.Var_idE.V
ar_idE,recordSym->stackOffset);
        insertCode(codeList, "\tpushl %%eax        \t#Push %%eax onto the stac
k\n");

```