

编译原理实验 3

课程：编译原理

211220169 祝明浩 211220182 刘钊瑜

专业：计算机科学与技术

一、 实验内容

已实现全部必做和选做 3.1 的内容, 能够对结构体变量进行相应处理与赋值, 并且能处理结构体变量作为函数参数的情况。

新增代码结构:

Inter_code.c/Inter_code.h:通过分析语法分析树以及在语义分析时填入的表项进行中间代码的翻译。

二、 编译方法

使用文件提供的 Makefile 文件, 执行 make 进行编译

三、 具体代码

为了能够处理结构体和数组类型的变量, 写了一个工具函数 getSize, 对于传入的类型 Type, 能够返回其在空间中所占的大小:

```

int getSize(Type type){
    if(type->kind==BASIC){
        return 4;
    }
    else if(type->kind==ARRAY){
        return type->u.array.size*getSize(type->u.array.elem);
    }
    else if(type->kind ==STRUCTURE){
        int size0=0;
        FieldList temp = type->u.structures.structure;
        while(temp!=NULL){
            size0+=getSize(temp->type);
            temp=temp->tail;
        }
        return size0;
    }
    return 0;
}

```

同时，在处理数组和结构体类型的变量时，有时候需要传递的是一个地址，而有时候需要传递的是这个地址上的值，因此在处理 Exp 时，trans_Exp 函数默认传回的是 Exp 的值，而 transaddr_Exp 函数默认传回的是 Exp 的值所在的地址：

```

void transaddr_Exp(Node* root,FILE*file,char* place){
    Node* n1=root->child;
    Node* n2 =n1 ->silbing;
    Node* n3 = NULL;
    if(n2!=NULL)
    {
        n3 = n2->silbing;
    }
    if(strcmp(n1->name,"ID")==0){
        FieldList newlist1=serchvarname(n1->sID);
        if(newlist1->inter==NULL)
            newlist1->inter=newvariable();
        if(place!=NULL){
            if(newlist1->flag1==0)
                fprintf(file,"%s := &%s\n",place,newlist1->inter);
            else
                fprintf(file,"%s := %s\n",place,newlist1->inter);
        }
    }
    else if(strcmp(n1->name,"Exp")==0){
        //| Exp LB Exp RB
        if(strcmp(n2->name,"LB")==0){
            char*t4=newtemp();
            transaddr_Exp(n1,file,t4);
            char*t1=newtemp();
            trans_Exp(n3,file,t1);
            char*t2=newtemp();
            Type type12 = Exp(n1);
            int sizenum=4;
            if(type12!=NULL && type12->kind==ARRAY){
                sizenum=getSize(type12->u.array.elem);
            }
            fprintf(file,"%s := %s * %d\n",t2,t1,sizenum);
            if(place!=NULL)
                fprintf(file,"%s := %s + %s\n",place,t4,t2);
        }
        else if(strcmp(n2->name,"DOT")==0){

```

四、 代码错误与解决方法

最开始由于简单的将数组都认为是 int 和 float 的数组, 没有考虑结构体为数组基础元素的情况, 导致寻址错误, 更改 getSize 和 transaddr_Exp 函数后, 能够正确寻址并且得到正确结果。