试设计Pascal记录变量（无变体）的属性翻译文法，并构造相应的语义动作程序。

Pascal记录变量的声明：

TYPE student = DATA

name: ARRAY[1…8] OF char;

sex: boolean;

age: integer;

END;

输入文法：

<data>→<entity>= DATA<list>END

<list>→<variable><variables>

<variables>→;<variable><variables> | ε

<variable>→<name>:<type>

属性翻译文法：

<data>→<entity>↑n= DATA↑k@init↑m, s, a<list>↓a, m, s ↑m', s' END @insertsym↓n, k, m', s'

<list>↓a, m, s ↑m', s' →<variable>↓a ↑L, a' @compinfo↓m, L ↑m', s'<variables>↓a', m', s' ↑m”,s”

<variables>↓a, m, s ↑m', s' →;<variable>↓a ↑L, a' @compinfo↓m, L ↑m', s' <variables> | ε

<variable>↓a ↑L, a'→<name>↑c\_n:<type>↑t, L @insertcomp ↓a, c\_n, t, L ↑a'

n= 记录名

k= 记录标识

m= 分量个数

a= 记录份量的符号表入口地址

s= 各分量大小之和

l= 分量大小

t= 分量类型

@init /\*初始化\*/

procedure init;

m:= 0; /\*分量个数清0\*/

s:= 0; /\*分量大小清0\*/

a:= NewEntry； /\*可用表项的入口地址\*/

end;

@insertsym↓n,k,m,s

/\*把记录名，记录标识，分量个数，各分量大小之和记录所需要的空间大小填入符号表\*/

Procedure insertsym(n, k, m, s)

@compinfo ↓m, L↑m', s'

Procedure compinfo(m, L)

m:= m+1; /\*分量个数 +1\*/

s:= s+L; /\*分量大小统计\*/

return(m, s);

end

@insertcomp ↓a, c\_n, t, L ↑a'

procedure insertcomp(a, c\_n, t, L)

/\*在a所指示的符号表项填入分量名字，类型和大小\*/

a= a+1;

return a; //将下一个可用表项的入口返回

end;