

1	Start	→ Stmt _{ast} \$ return (ast)	(13)
2	Stmt _{result}	→ id _{var} assign E _{expr} result ← MAKEFAMILY(assign, var, expr)	(14)
3		if lparen E _p rpren Stmt _s fi result ← MAKEFAMILY(if, p, s, MAKENODE())	(15)
4		if lparen E _p rpren Stmt _{s1} else Stmt _{s2} fi result ← MAKEFAMILY(if, p, s1, s2)	(16)
5		while lparen E _p rpren do Stmt _s od result ← MAKEFAMILY(while, p, s)	(17)
6		begin Stmts _{list} end result ← MAKEFAMILY(block, list)	(18)
7	Stmts _{result}	→ Stmts _{sofar} semi Stmt _{next} result ←sofar.MAKE SIBLINGS(next)	(19)
8		Stmt _{first} result ← first	(20)
9	E _{result}	→ E _{e1} plus T _{e2} result ← MAKEFAMILY(plus, e1, e2)	(21)
10		T _e result ← e	(22)
11	T _{result}	→ id _{var} result ← MAKENODE(var)	(23)
12		num _{val} result ← MAKENODE(val)	(24)

Figure 7.17: Semantic actions for grammar in Figure 7.14.