**First Sets**

**Eric Dodds**

<program> => FIRST(PLATYPUS)

<statements> => FIRST(AVID, SVID, IF, USING, INPUT, OUTPUT)

<statements’> => FIRST(AVID, SVID, IF, USING, INPUT, OUTPUT, ε)

<statement> => FIRST(AVID, SVID, IF, USING, INPUT, OUTPUT)

<assignment statement> => FIRST(AVID, SVID)

<assignment expression> => FIRST(AVID, SVID)

<selection statement> => FIRST(IF)

<opt\_statements> => FIRST(AVID, SVID, IF, USING, INPUT, OUTPUT, ε)

<iteration statement> => FIRST(USING)

<input statement> => FIRST(INPUT)

<variable list> => FIRST(AVID\_T, SVID\_T)

<variable list’> => FIRST(AVID\_T, SVID\_T, ε)

<output statement> => FIRST(OUTPUT)

<opt\_variable list> => FIRST(AVID\_T, SVID\_T, ε)

<arithmetic expression> => FIRST( + , - )

<unary arithmetic expression> => FIRST( + , - )

<additive arithmetic expression> => FIRST( + , - )

<additive arithmetic expression’> => FIRST( + , - , ε)

<multiplicative arithmetic expression> => FIRST( \* , / )

<multiplicative arithmetic expression> => FIRST( \* , / , ε)

<primary arithmetic expression> => FIRST(AVID\_T, FPL\_T, INL\_T, + , - )

**3.3.2**

<string expression> => FIRST(SVID\_T, STR\_T)

<string expression’> => FIRST( <> , ε )

<primary string expression> => FIRST(SVID\_T, STR\_T)

**3.3.3**

<conditional expression> => FIRST(OR, AND)

<logical OR expression> => FIRST(OR)

<logical OR expression’> => FIRST(OR, ε)

<logical AND expression> => FIRST(AND)

<logical AND expression’> => FIRST(AND, ε)

LF

<logical OR expression> -> <logical AND expression><logical OR expression’>

<logical OR expression’>-> .OR. <logical AND expression> <logical OR expression’>| empty

<logical AND expression> -> <logical AND expression’> .AND. <relational expression> | <relational expression>

LF

<logical AND expression> -> <relational expression><logical AND expression’>

<logical AND expression’> -> .AND. <relational expression><logical AND expression’> | ε

**3.3.4**

<relational expression> => FIRST(AVID\_T, FPL\_T, INL\_T, SVID\_T, STR\_T, )

<relational expression op> => FIRST(==, !=, <, > )

<primary a\_relational expression> => FIRST(AVID\_T, FPL\_T, INL\_T)

<primary s\_relational expression> => FIRST(SVID\_T, STR\_T)