Mitchell Cummins

Test Plan

Test Case 1:

Code:

var

integer x;

integer m;

begin

x = 6;

m = 0;

while (m<x) loop

output(“the value of m is: ”)

output (m)

end loop

end

Symbol Table:  
t\_integer : x

t\_integer : m

token/lexems:

t\_var : var

t\_integer : integer

t\_id : x

t\_integer : integer

t\_id : m

t\_begin : begin

t\_id : x

t\_assign : =

t\_number : 6

t\_id : m

t\_assign : =

t\_number : 0

t\_while : while

s\_lparen : (

t\_ id : m

s\_lt : <

t\_id : x

s\_rpren : )

t\_loop : loop

t\_output : output

s\_lparen : (

t\_string : “the value of m is: ”

s\_rpren : )

t\_output : output

s\_lparen : (

t\_id : m

s\_rpren : )

t\_end : end

t\_loop : loop

t\_end : end

Test Case 2:

Code:

var

integer x;

begin

x = 6;

output(“the value of x is: “)

output(x)

end

Symbol Table:  
t\_integer : x

token/lexems:

t\_var : var

t\_integer : integer

t\_id : x

t\_integer : integer

t\_id : m

t\_begin : begin

t\_id : x

t\_assign : =

t\_number : 6

t\_id : m

t\_assign : =

t\_number : 0

t\_while : while

s\_lparen : (

t\_ id : m

s\_lt : <

t\_id : x

s\_rpren : )

t\_loop : loop

t\_output : output

s\_lparen : (

t\_string : “the value of m is: ”

Test Case 3:

Code:

var

integer x;

begin

x = 1;

while(x!=10 or x == false) loop

output(“the value of x is: “)

output(x)

x + 1;

end loop

end

Symbol Table:  
t\_integer : x

token/lexems:

t\_var : var

t\_integer : integer

t\_id : x

t\_begin : begin

t\_id : x

t\_assign : =

t\_number : 1

t\_while : while

s\_lparen : (

t\_ id : x

s\_ne : !=

t\_number : 10

t\_or : or

t\_ id : x

s\_eq : ==

t\_false : false

s\_rpren : )

t\_loop : loop

t\_output : output

s\_lparen : (

t\_string : “the value of x is: ”

s\_rpren : )

t\_output : output

s\_lparen : (

t\_id : x

s\_rpren : )  
t\_id : x

s\_plus : +

t\_number : 1

t\_end : end

t\_loop : loop

t\_end : end