

```

function simple_tree_matching(A, B)
if the roots of the two trees A and B contain distinct symbols, then
return (0)
m := the number of the first level subtrees of A
n := the number of the first level subtrees of B
Initialization M [i,0] := 0 for i=0, ..., m, M[0,j]:= 0 for j=0,...,n

for i:= 1 to m do
  for j:= 1 to n do
    M[i, j] = max (M[i, j-1], M[i-1, j] M[i-1, j-1]+W[i, j])
    where W[i, j] = simple_tree_matching (A_i, B_j) where A_i
and B_j are the ith and jth first level subtrees of A and B
  end for
end for
return M[m,n]+1

```

Past		Current	
p0	mA (){	c0	mA (){
p1	if (pred_a) {	c1	if (pred_a0) {
p2	foo()	c2	if (pred_a) {
p3	}	c3	foo()
p4	}	c4	}
p5	mB (b) {	c5	}
p6	a := 1	c6	}
p7	b := b+1	c7	mB (b) {
p8	fun (a,b)	c8	b := b+1
p9	}	c9	a := 1
		c10	fun (a,b)
		c11	}