1. evaluate new Class1(5,6) (its value is b1)

Class1: 1 2 3 4   b1

yp __ 6 __   z __ 11__

xp __ 5 __

active frame before the call

b1

x __ 5 ___   getX()   Class1

y __ 11 ___   max()   Class1(**int**, **int**)
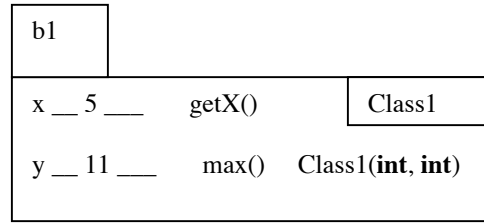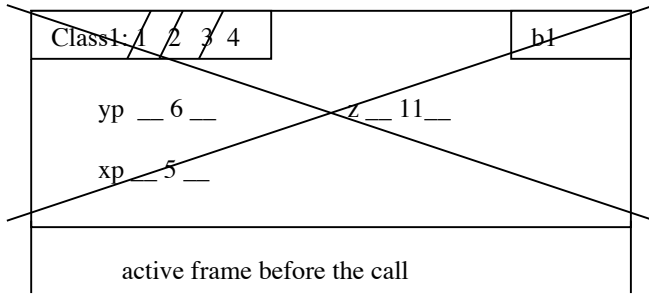
2. evaluate Class1.max(6,5). We show the file drawer because it has a static components that are needed.

max: 1   Class1

y __ 5 __

x __ 6 __

active frame before the call

ZERO __ 0 __

max(**int**, **int**)

(plus any folders for Class1)

file drawer for Class1
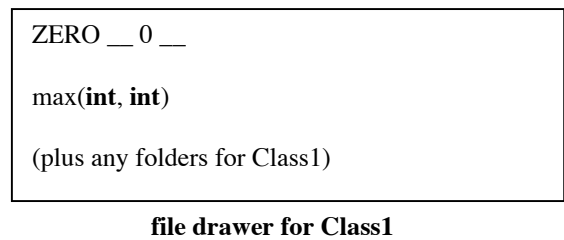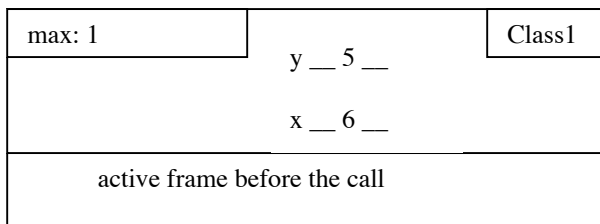
3. Evaluate d.max(d.getX(), e.getX()). This caused some confusion, because the instance name d is used to access static method max. This works, because of the inside-out rule. Note that argument values are pushed on the call stack first. So, after this has been done, the call stack looks as shown to the left at the bottom, where we have drawn an X through each erased frame. To the right at the bottom, we show the stack without the two erased frames and after the call on d.max has completed, with its value still on the call stack. And, we show the file drawer for Class1.

ZERO __ 0 __   max(**int**, **int**)

a0

x __ 8 ___   getX()   Class1

y __ 3 ___   max()   Class1(**int**, **int**)

a1

x __ 2 ___   getX()   Class1

y __ 4 ___   max()   Class1(**int**, **int**)

file drawer for class Class1

2

getX: 1   a1

8

getX: 1   a0

active frame before the first call

8

max: 1   Class1

y __ 2 __

x __ 8 __

active frame before the first call
d ___ a0 __   e ___ a1 ___