Method Invocation, Stack Frames

When you call a method, what happens?

1. Space is created on the stack for params, local vars
   “stack frame” or “activation record”

2. Parameter values are copied onto the stack frame call-by-value

```java
boolean foo (int a, int b) { double x = -- ; }
```

---

call:  $foo(2+4, 9)$
Rectangle longerDiagonal(Rectangle a, Rectangle b) {
    double aDiag = diagonal(a);
    double bDiag = diagonal(b);
    if (aDiag > bDiag) return a; else return b;
}

double diagonal(Rectangle r) {
    return Math.sqrt(square(r.width) + square(r.height));
}

double square(double x) {
    return x * x;
}

// from the class Math:
double sqrt(double d) \( \Rightarrow \sqrt{d} \)
The Heap:
address 100  address 182
3
4
6

The Stack over time:

b
Diag
a
Diag
b
a
ret
182
100

Call: longerDiagonal(a,b)

b
Diag
a
Diag
b
a
ret
182
100

Call: diagonal(a)

x
ret
100

Call: Square(3)
return from: Square(3)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)

intermediate
ret
bDiag
aDiag
b
a
ret
182
100

return from: Square(4)
call: \texttt{sqrt(25)}

return from: \texttt{sqrt(25)}

return from: diagonal(a)
We continue the same way as before, calling diagonal(b), so after that runs and the stack grows and shrinks as before, we finally...

```
ret 10

b
bDiag 10
|
|
5
a
aDiag 182
|
|
100
|
|
ret 182

b
182
|
|
ret 182

a
182
|
|
ret 182

b
182
|
|
100
|
|
return from: diagonal(b)

return from: longerDiagonal(a,b)
```

A reference to rectangle \( \Box \) is returned.
Some notes:

1. Parameter passing is **call-by-value** — the values of the parameters are copied to the stack frame of the call.

2. Stack frames contain references to objects on the heap — any object accessible (directly or indirectly) from the stack is NOT garbage.

3. Each invocation of a method (each time you call it), a new stack frame is created.