

CS100J 18 September 2003

More on Methods. Scope of parameters. Executing a call. Conditional statements. Blocks. The return statements. Read section 2.3 for this lecture and the next.

Research on spelling

According to a research at Cambridge University, it doesn't matter in what order the letters in a word are, the only important thing is that the first and last letter be at the right place. The rest can be a total mess and you can still read it without problem. This is because the human mind does not read every letter by itself, but the word as a whole.

Last time: Saw that the body of a procedure is a sequence of statements, which are executed in order when the method is called

```
/** Print b, c, and their sum */  
public static void print(int b, int c) {  
    System.out.println( b);  
    System.out.println(c);  
    System.out.println(b + c);  
}
```

```
/** Print b, c, and their sum */  
public static void print(int b, int c) {  
    System.out.println( b);  
    System.out.println(c);  
    System.out.println(b + c);  
}
```

Parameters b and c are *variables*. They are created when the method is called and destroyed when the method call is finished.

The *scope of a parameter* --the places where it can be referenced or used, is the method body itself.

```
/** Print b */  
public static void print (int b) {  
    System.out.println(b);  
}
```

```
/** Print b, c, and their sum */  
public static void print(int b, int c) {  
    System.out.println( b);  
    System.out.println(c);  
    System.out.println(b + c);  
}
```

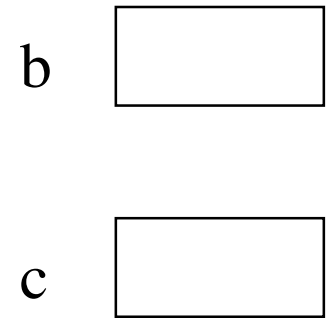
How is a call like this executed? print(5, 6)?

- Step 1: create the parameters (variables).
- Step 2: assign the arguments to the parameters.
- Step 3: execute the method body.
- Step 4: erase the parameters.

```
/** Print b, c, and their sum */  
public static void print(int b, int c) {  
    System.out.println( b);  
    System.out.println(c);  
    System.out.println(b + c);  
}
```

How is a call like this executed? print(5, 6)?

- Step 1: create the parameters (variables).
- Step 2: assign the arguments to the parameters.
- Step 3: execute the method body.
- Step 4: erase the parameters.



A method can call another method

```
/** Print b, c, and the sum of their squares */  
public static void print(int b, int c) {  
    System.out.println( b);  
    System.out.println(c);  
    printSum(b*b, c*c);  
}
```

```
/** Print x + y */  
public static void printSum(int x, int y) {  
    System.out.println(x+y);  
}
```