

Control in Java and Programming

- The computer executes your program one statement at a time, from top to bottom within the method/function that's called.

```
public int daysIn(int month) {  
    if (month == 1) return 31;  
    if (month == 2) return 28;  
    if (month == 3) return 31;  
    ...  
}
```

- What is the anatomy of a function? Of an if-statement? How do we specify these?

Lydia Kavradi

- Awards
 - > Grace Murray Hopper
 - > Brilliant 10

"I like to work on problems that will generally improve the quality of our life,"

What's the thing you love most about science?

"Working with students and interacting with people from diverse intellectual backgrounds. Discovery and the challenge of solving a tough problem, especially when it can really affect the quality of our lives. I find the whole process energizing."

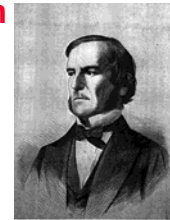


Specifying language constructs

```
public Type Name(parameter-list) {  
    statement-list  
}  
  
parameter-list:: empty-list || non-empty-list  
non-empty-list:: Type Name  
non-empty-list:: Type Name, non-empty-list  
  
if ( boolean-expression ) {  
    statement-list  
}
```

zero-one, true-false, boolean

```
if ( a == b ) {  
    statement-list  
}  
  
if ( a relational-operator b ) {  
    statement-list  
}  
  
if ( bool-exper conditional bool-expr ) {...}  
if ( boolean-function ) {...}  
  
==, !=, <, >, <=, >=  
&& ||, !
```



John Kemeny, (1926-1992)

- Invented BASIC, assistant to Einstein, Professor and President of Dartmouth
 - Popularized computers being ubiquitous on campus/at home
 - BASIC ported to early personal computers by Gates and Allen
- Initially BASIC was free, but many different dialects arose. In 1985 Kemeny and Kurtz shipped TRUE BASIC, to challenge Pascal in academia
 - What's used today?



Equality of values and objects

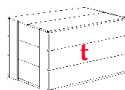
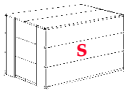
```
int x = 3*12;
if (x == 36) {is-executed}
String s = new String("genetic");
String t = s.substring(0,4);
if (t == "gene") {not executed}
if (t.equals("gene")) {is-executed}
```

- Primitive types are boxes
- Object types are labels on boxes
 - If we don't call new there's no box for the label
 - No box is called *null*, it means no object referred to or referenced by variable/pointer/reference

Objects and values

- Primitive variables are boxes
 - think memory location with value
- Object variables are labels that are put on boxes

```
String s = new String("genome");
String t = new String("genome");
if (s == t) {they label the same box}
if (s.equals(t)) {contents of boxes the same}
```



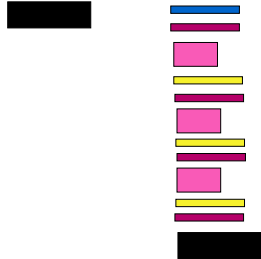
What's in the boxes? "genome" is in the boxes

Objects, values, classes

- For primitive types: int, char, double, boolean
 - Variables have names and are themselves boxes (metaphorically)
 - Two int variables assigned 17 are equal with ==
- For object types: String, Sequence, others
 - Variables have names and are labels for boxes
 - If no box assigned, created, then label applied to *null*
 - Can assign label to existing box (via another label)
 - Can create new box using new
- Object types are references or pointers or labels to storage

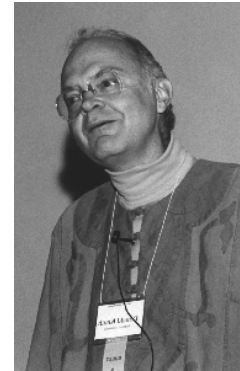
Anatomy of for-loop

```
String s = new String("agt");  
String rs = new String("");  
for (int k=0; k < 3; k++){  
    rs = rs + s.charAt(k);  
}
```



- Initialization happens once
- Loop test evaluated
 - If true body executes
 - If false skip after loop
- After loop body, increment executed and test re-evaluated
- What should be true about test?
- What about body?
- What about together?

Don Knuth (Art of Programming)



"My feeling is that when we prepare a program, it can be like composing poetry or music; as Andrei Ershov has said, programming can give us both intellectual and emotional satisfaction, because it is a real achievement to master complexity and to establish a system of consistent rules."

"We have seen that computer programming is an art, because it applies accumulated knowledge to the world, because it requires skill and ingenuity, and especially because it produces objects of beauty."