CMSC 131

Fall 2018

Announcements

- The deadline for First project (Hello World) has been extended to 9/12. If you're stuck, come to office hours

Vague Definitions (for now…)

First, let's loosely define these terms:

- Object
- Class
- Method
  - Main method
- Statements

Example: SimpleExample.java

Things to observe:

- We are looking at a class called "SimpleProgram"
- There is just one method, called main
- The main method contains a few statements
- There are two kinds of "comments"
- Most statements end with semi-colons
- System.out.print is a primitive tool for text output
  - Note the difference in print vs. println

Example: VariablesExample.java

Things to observe:

- Two "local" variables are declared (their type is "int")
- The "assignment operator" stores values inside the variables
- Memory diagram (at the end):

socksLostInDryer | 9
numberOfSocks | 18

Call Stack

Variable Types

- Primitives (basic "atoms")

- References to Objects (later)

Primitive Types (Whole Numbers First)

- An `int` variable takes up 4 bytes of memory. What range of values can be stored?

Type	Memory used	Range of values that can be stored
`long`	8 bytes	`-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807`
`int`	4 bytes	`-2,147,483,648 to 2,147,483,647`
`short`	2 bytes	`-32,768 to 32,767`
`byte`	1 byte	`-128 to 127`

- Examples:

```
int secondsPerYear = 31557600;

long humansOnEarth = 7625913792L;   // note the L

byte age = 19;
```

Primitive Types (Floating Point Values)

- Note: Most real numbers cannot be represented! Why not?

Type	Memory used	Range and precision (for positive values)
double	8 bytes	3.40282347 x , 1.40239846 x
float	4 bytes	1.7976931348623157 x , 4.9406564584124654 x

- Examples:

```
double velocity = 325.92732;

float distance = 52.25F;      // note the F
```

Primitive Types (Characters and Boolean)

- char

Example:

```
char letterGrade = 'A';     // Note the single quotes
```


- boolean

Examples:

```
boolean result = true;
boolean state = false;
boolean hasFever = (temperature > 98.6);
```