Review 1

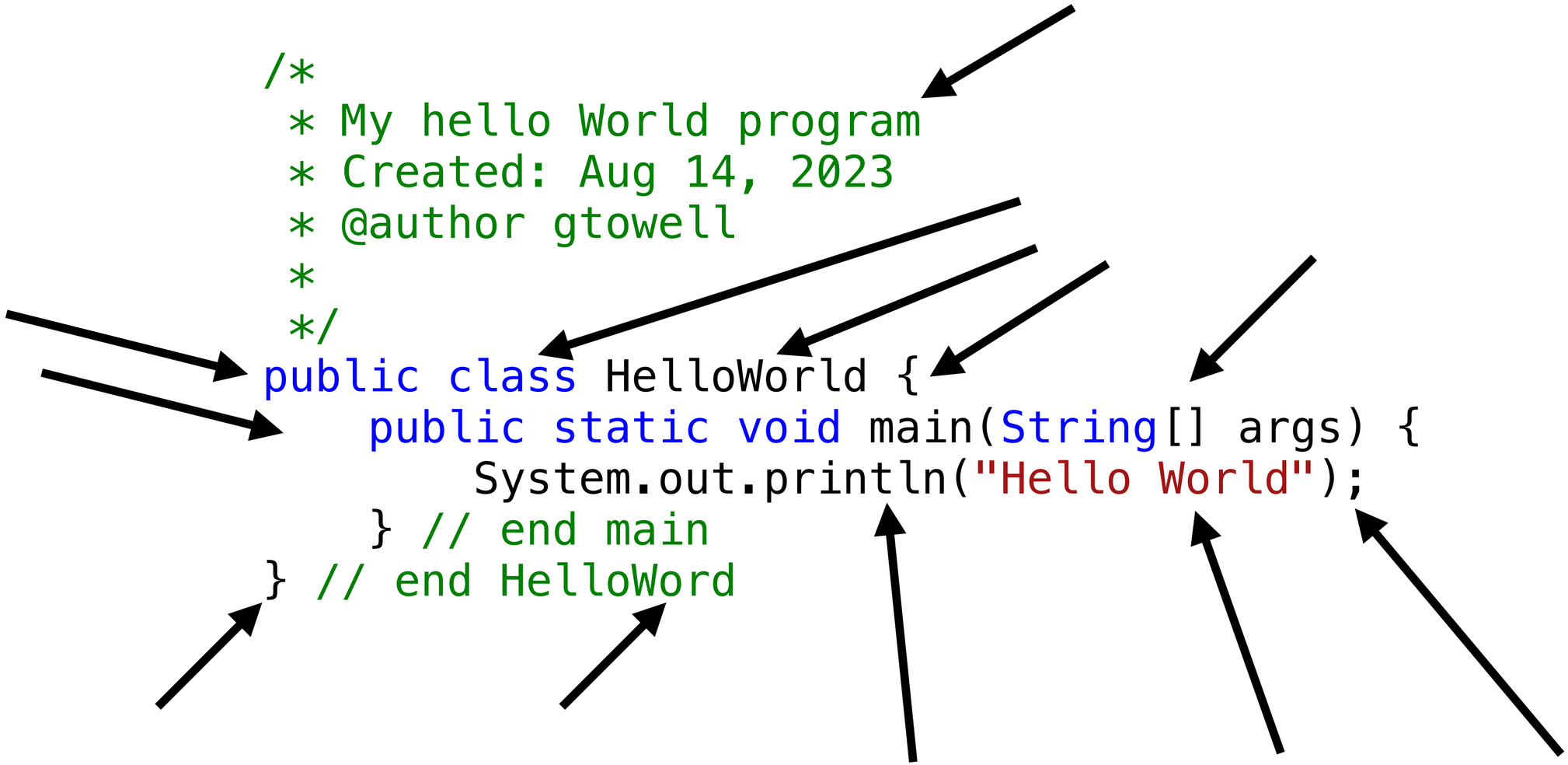
CS113, Oct 2

Algorithms

- Algorithms are a precise statement of how to solve a problem
  - NOT a program
  - NOT written using a PL

  - Write in a way that is easy for you
  - Use a pencil
    - draw circles and arrows
  - Be very precise

Hello World



```java
/*
 * My hello World program
 * Created: Aug 14, 2023
 * @author gtowell
 *
 */
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello World");
    } // end main
} // end HelloWord
```

Variables

- A thing in a program that holds a value

- Declaration -- the name and type of the variable
  - `String aString;`
  - Variables must be "declared" before they can be used
    - declaration occurs once
    - `double d = 5.0;  // declare and assign on same line`
- Assignment
  - `aString = "aaa";`
    - assignment my occur many times
- Read
  - Use the value that was initialized / assigned
  - variables must be initialized (assigned) before they can be read

Data Types

- Integer aka int
  - int base = 55; //32 bits 00000000 00000000 00000000 00110111
- String -- a sequence of chars
  - String base = "55"; //16 bits 00110101 00110101
- double base = 55.0; //64 bits
  1000000010010111000000000000000000000000000000000000000000000000
- boolean b = true; // or false // 1 bit

Casting

easy / standard converting one type to another

- Often Java will convert types for you
- Sometimes, you need to tell Java exactly what you want
  - this is called "Casting"
- For example, the code at right fails to compile with the message

```
Casting1.java:6: error: incompatible types: possible
lossy conversion from double to int
        int resultInt = aDouble * anInt;
```

- To fix, cast
  - change line to (int)(aDouble*anInt);

Harder conversions

- String to int
- String to double
- degrees to radians
- number, power to number

if

- if (boolean) { do something }
- else { do something else }

- Example:
  - Suppose Java did not have the modulus (remainder) operator, %
  - Given two numbers from the command line, print true if the remainder of the first with respect to the second is 0. Otherwise print the remainder.

If

No modulus operator

```
public static void main(String[] args) {
        int num = Integer.parseInt(args[0]);
        int den = Integer.parseInt(args[1]);
        int div = num / den; // this will round down so 5/2 = 2
        int mul = div * den;
        int modu = num - mul;
        if (modu == 0) {
           System.out.println("true");
        } else {
            System.out.println("The remainder of " + num + "/" + den + " is " + modu);
        }
    }
```

Java Random number generator

- Random double in range 0.0 .. <1.0
  - double d = Math.random()

- Random double in range 0.0 .. <100.0
  - double d = Math.random()*100

- Random double in range lo .. < hi
  - double d = (Math.random()*(hi - lo)) + lo //

- Random integer in range lo .. < hi
  - int i = (int)((Math.random()*(hi - lo)) + lo)

While loops

- ```
  while (boolean is true) {
        do something
  }
  ```

Count the number of times must draw a random number in the range 20-100 to get a number greater than 90
  ```

While loop

Count to > 90

```
public static void main(String[] args) {
    int count = 0;
    int drawn = 0;
    while (drawn < 90) {
        count++;
        drawn = (int) (Math.random() * (100 - 20 + 1) + 20);
        // to get a number in the range 20-100 including both 20 and 100 you need 81 possibilities,
        // not just 80
    }
    System.out.println(count);
}
```

"For" loop

for (**INIT** ; **CONDITION** ; **UPDATE**) {
    BODY
}

For-loops are preferred when you have the init/condition/body/update pattern, and/or when you know how many times you want the thing to run

Usually update is increment by 1

```
for (int i=0; i<100; i++)
```

BUT update can be decrement by 1

```
for (int i=100; i>=0; i--)
```

OR any amount

```
for (int i=0; i<100; i += 7)
```

OR even multiplication/division

```
for (int i=1; i<1000; i=i*2)
```

Fizz Buzz:
print each of the numbers in 0 .. 100 except
if num evenly divisible by 3 print "fizz"
if num evenly divisible by 5 print "buzz"
is both 3 and 5 print "fizz buzz"

For loop

Fizz Buzz

```
public static void main(String[] args) {
        for (int i = 0; i <= 100; i++) {
            if (i % 3 == 0 && i % 5 == 0) {
                System.out.println("Fizz Buzz");
            } else if (i % 3 == 0) {
                System.out.println("Fizz");
            } else if (i % 5 == 0) {
                System.out.println("Buzz");
            } else {
                System.out.println(i);
            }
        }
    }
```

Loops in loops

- First, given a number, x, is x prime?

- Find/print all primes less than 100?
  - This will require nested loops!

IsPrime

```
public static void main(String[] args) {
        Integer num = Integer.parseInt(args[0]);
        boolean isPrime = true; // start by assuming the number is prime
        for (int i = 2; i < num; i++) {
            if (num % i == 0) {
                isPrime = false;
            }
        }
        if (isPrime) {
            System.out.println(num + " is prime");
        } else {
            System.out.println(num + " is NOT prime");

        }
    }
```

Primes to 100

using much of isPrime!

Actually this finds primes up to a user supplied number

```
public static void main(String[] args) {
        Integer num = Integer.parseInt(args[0]);
        for (int j = 2; j < num; j++) {
            boolean isPrime = true; // start by assuming the number is prime
            for (int i = 2; i < j; i++) {
                if (j % i == 0) {
                    isPrime = false;
                }
            }
            if (isPrime) {
                System.out.println(j);
            }
        }
    }
```