CSC 212 PROGRAMMING WITH DATA STRUCTURES

SPRING 2016

PROF. SARA SHEEHAN

SMITH COLLEGE

CLASS 1: JAN 26 OUTLINE

- **Introduction**

- **Course overview and goals**

- **Introduction to Java**

- **Java to Python reverse engineering**

- **Syllabus**

- **(if time) First data structure**

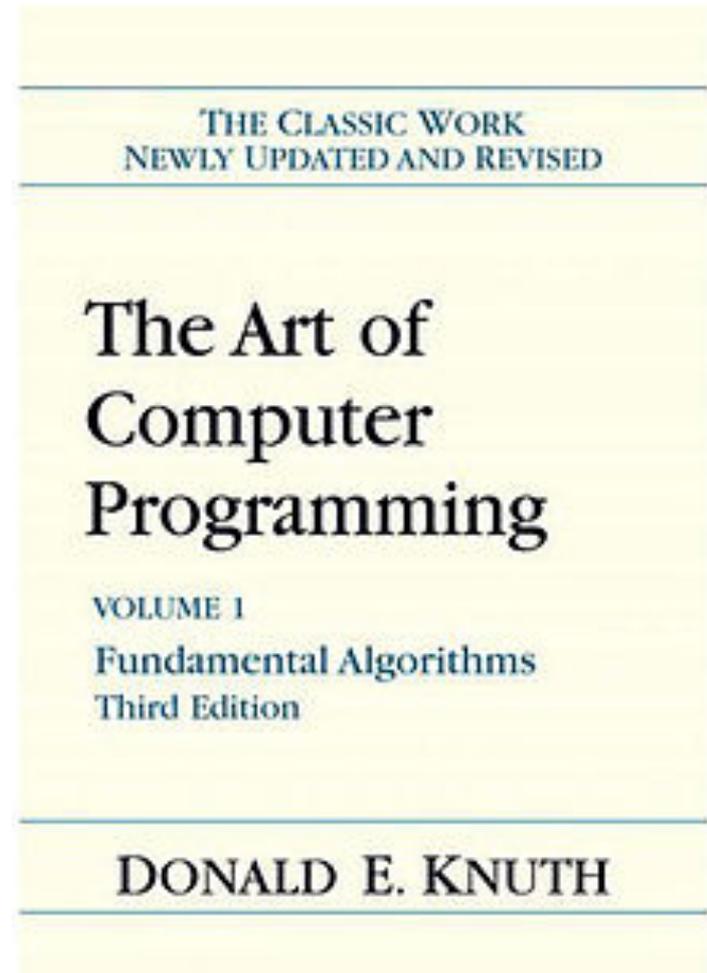# FOOD FOR THOUGHT

THE ART OF PROGRAMMING

With a partner:

1) Names (be ready to introduce your partner to the class!)

2) What you did for winter break

3) Brainstorm how or why programming might be considered an art

4) Example of software you have used that you would consider artistic

THE CLASSIC WORK
NEWLY UPDATED AND REVISED

The Art of
Computer
Programming

VOLUME 1
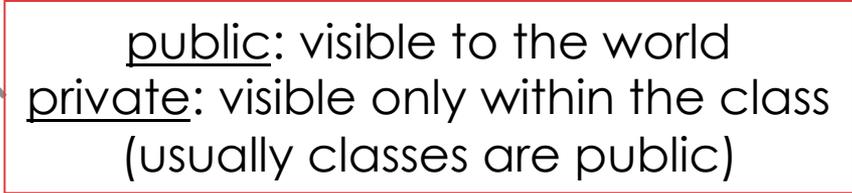Fundamental Algorithms
Third Edition

DONALD E. KNUTH

COURSE GOALS

- **Essential Data Structures and Algorithms**

- **Java Programming Language**

- **Ability to choose an effective data structure for a new task**

- **Relating theory and programming details**

- **Confidence to learn a new language in the future**

- **Coding style and workflow practices**

BUILDING A JAVA PROGRAM

public class AddTax {

public: visible to the world
private: visible only within the class
(usually classes are public)

}

BUILDING A JAVA PROGRAM

public class AddTax {

        public static final double *TAXRATE = 0.0625;*

static: makes this field global

}

BUILDING A JAVA PROGRAM

public class AddTax {

        public static final double *TAXRATE = 0.0625;*

static: makes this field global

final: makes this field immutable

}

BUILDING A JAVA PROGRAM

```
public class AddTax {

        public static final double TAXRATE = 0.0625;
```

double: type of the field, you must declare the type!

static: makes this field global

final: makes this field immutable

```
}
```

BUILDING A JAVA PROGRAM

```java
public class AddTax {

        public static final double TAXRATE = 0.0625;


        private static double addTax(double stickerPrice) {

                double finalCost = (1 + TAXRATE) * stickerPrice;

                return finalCost;

        }



}
```

argument(s): like python, but you must declare their type

BUILDING A JAVA PROGRAM

```java
public class AddTax {

    public static final double TAXRATE = 0.0625;



    private static double addTax(double stickerPrice) {

        double finalCost = (1 + TAXRATE) * stickerPrice;

        return finalCost;

    }



}
```

argument(s): like python, but you must declare their type

return type: methods can either return a type or return "void"

BUILDING A JAVA PROGRAM

```java
public class AddTax {

        public static final double TAXRATE = 0.0625;

        private static BufferedReader stdin =

                new BufferedReader(new InputStreamReader(System.in));

        private static double addTax(double stickerPrice) {

                double finalCost = (1 + TAXRATE) * stickerPrice;

                return finalCost;

        }

        public void main(String[] args) throws IOException {



        }

}
```

return type for main is void

BUILDING A JAVA PROGRAM

```java
public class AddTax {

        public static final double TAXRATE = 0.0625;

        private static BufferedReader stdin =

                new BufferedReader(new InputStreamReader(System.in));

        private static double addTax(double stickerPrice) {

                double finalCost = (1 + TAXRATE) * stickerPrice;

                return finalCost;

        }

        public void main(String[] args) throws IOException {

                System.out.print("Please enter the price:  ");

                String line = stdin.readLine();

                double stickerPrice = Double.parseDouble(line);



        }

}
```

return type for main is void

convert a string to a number

BUILDING A JAVA PROGRAM

```java
public class AddTax {

        public static final double TAXRATE = 0.0625;

        private static BufferedReader stdin =

                new BufferedReader(new InputStreamReader(System.in));

        private static double addTax(double stickerPrice) {

                double finalCost = (1 + TAXRATE) * stickerPrice;

                return finalCost;

        }

        public void main(String[] args) throws IOException {

                System.out.print("Please enter the price:  ");

                String line = stdin.readLine();

                double stickerPrice = Double.parseDouble(line);

                double finalPrice = addTax(stickerPrice);

                System.out.println("With tax, that comes to $" + finalPrice + ".");

        }

}
```

return type for main is void

convert a string to a number

printing

BUILDING A JAVA PROGRAM

```java
public class AddTax {

        public static final double TAXRATE = 0.0625;

        private static BufferedReader stdin =

                new BufferedReader(new InputStreamReader(System.in));

        private static double addTax(double stickerPrice) {

                double finalCost = (1 + TAXRATE) * stickerPrice;

                return finalCost;

        }

        public void main(String[] args) throws IOException {

                System.out.print("Please enter the price:  ");

                String line = stdin.readLine();

                double stickerPrice = Double.parseDouble(line);

                double finalPrice = addTax(stickerPrice);

                System.out.println("With tax, that comes to $" + finalPrice + ".");

        }

}
```

COMMON JAVA ERRORS

- **Missing bracket**

- **No type specifier**

- **Omit static on a field or main**

- **Argument/parameter type mismatch**

- **Field/method out of place**

- **No class**

(Compiler vs. Runtime Error)

Demo: AddTax in Java (Eclipse and Commandline)

GROUP ACTIVITY: REVERSE ENGINEERING

In groups of 2-3, write down
what this same program
would look like in Python.