Lesson 4-1: Numbers, Arithmetic, and I/O

Computer Science 46A: Introduction to Programming

San José State University

Announcements

- Homework #4(aka HW2) is due 2/24
- Lab #4 is this Friday (2/21)
- Please stay till end of class. Don’t leave before 1:15pm
  - Make best use of LA when you can

Pre-Class Readings

4.1 Numbers

4.2 Arithmetic

4.3.1 Input and Output: Reading Input

Learning Objectives

By the end of this lesson, you should be able to:

1. Use a Scanner object to take user inputs into a class
2. Perform simple arithmetic with mathematical operators
3. Use methods from the Math class to perform more complex mathematical operations

I/O with Scanner Objects

The `Scanner` Class

- Java has many classes which are available to developers when creating new programs
- To input existing Java classes, add an import statement to the top of your script:

```
import java.[lang].[class]
```

- The first class we will explore is `java.util.Scanner`
- The Scanner class allows us to take input from the user while the program is running

Scanner objects

- We create Scanner objects the same we use other classes:

```
Scanner myObj = new Scanner(System.in);
```

- Then, we tell the Scanner object to look for different types of output from the user:

```
int intNum = myObj.nextInt();
double doubleNum = myObj.nextDouble();
```

Example: InputUsingScanner.java

```
public class InputUsingScanner
{
    public static void main(String[] args)
    {
        // System.in: the keyboard
        Scanner in = new Scanner(System.in);

        // Read an integer and store it in
        // an int variable
        int intNum = in.nextInt();

        // Calculate the square of intNum and store it in
        // an int variable square
        int square = intNum * intNum;

        // Display the square of intNum
        System.out.println("The square of " + intNum +
                           " is " + square + ".");

    }
}
```

A Scanner object allows for user input from the keyboard

This script is expecting an integer to be entered by the user

After the input is received, the rest of the code runs

Poll Everywhere: Question 1

What happens if you input a number with a decimal, e.g. 5.7?

A) The program will return “The square of 5.7 is 32.49”

B) The program will return “The square of 5.7 is 25”

C) The program will return `java.util.InputMismatchException`

Poll Everywhere: Question 2

What happens if you input 1000000 (one million)?

A) The program will return “The square of 1000000 is 1000000000000”

B) The program will return “The square of 1000000 is -727379968”

C) The program will return `java.util.InputMismatchException`

Numbers

Primitive Data Types: Numbers

Type	Description	Size (bytes)
byte	-128 . . . 127	1
short	-32,768 . . . 32,767	2
int	-2,147,483,648 . . . 2,147,483,647	4
long	$-9{,}223{,}372{,}036{,}854{,}776 \times 10^{18}$ . . . $9{,}223{,}372{,}036{,}854{,}776 \times 10^{18}$	8
float	A range of $\pm 10^{38}$ and about 7 significant decimal digits	4
double	A range of $\pm 10^{308}$ and about 15 significant decimal digits	8

No Decimals

Decimals

Note: A byte is 8 bits. A bit is either a 0 or a 1. Numbers are stored in binary format in powers of 2

Knowing the type of your number is key!

- In `InputUsingScanner.java`, we can calculate the square of 1 million by changing the `int` to a `long`
- You can change the type of any number using the syntax

```
(new type) [var name]
```

- For example: if `intNum` is an `int` assigned 1000000, we change it to a long number using

```
(long) intNum
```

- Beware, changing numbers to types with lower bytes will result in a loss of information
  - E.g. You can change an `int` to a `double` without losing any information
  - E.g. If you change a `double` to an `int`, you will lose all of the decimals!

Arithmetic

Arithmetic Operators

Arithmetic operators are the basic math operations we are familiar with

+ Addition
- Subtraction
* Multiplication
/ Division
% Remainder (used for integers)

Example: OperatorPreference.java

```java
public class OperatorPrecedence
{
    public static void main(String[] args)
    {
        // System.in: the keyboard
        Scanner in = new Scanner(System.in);

        // Display an input prompt to get three integers
        System.out.print("Enter three integers: ");

        // Read three integers and store them in
        // three int variables num1, num2, and num3
        int num1 = in.nextInt();
        int num2 = in.nextInt();
        int num3 = in.nextInt();

        int result = num1 / num2 * num3;

        // Display the result
        System.out.println("The result is " + result + ".");
    }
}
```

A Scanner object is used to read in three integers

After three inputs, the operation is carried out

Poll Everywhere: Question 3

```
result = num1  / num2 * num3
```

What happens if you input 12, 3, and 4?

A) The program will return “The result is 16.”

B) The program will return “The result is 1.”

C) The program will return `java.util.InputMismatchException`

Order of Operations

- Arithmetic operators follow the same steps as we are taught in math, working left to right

1. Parentheses
2. Exponents
3. Multiplication and Division
4. Addition and Subtraction

Did you learn: “Please Excuse My Dear Aunt Sally”?

Poll Everywhere: Question 4

```
result = num1 + num2 * num3
```

What happens if we change the first operator to a plus and you input 12, 3, and 4?

A) The program will return “The result is 60.”

B) The program will return “The result is 24.”

C) The program will return `java.util.InputMismatchException`

Going beyond arithmetic

- What if we want to do more complicated calculations that simple arithmetic?
- Java's Answer: The `java.lang.Math` class
- Some examples of the Math class
  - Exponents and logarithms
  - Classic trigonometry functions
  - Minimums, maximums
- Example: `MathExamples`

Participation Exercise 4-1a: AverageOfThreeIntegers

Goal: Write a program to display the sum and average of three integers

```
Enter the first integer: 2
Enter the second integer: 5
Enter the second integer: 3
The sum of the three integers is 10.
The average of the three integers is 3.3333333333333335.
```

Results of AverageOfThreeIntegers.java

Hint: Given integers, how will you ensure your average is calculated with decimal places?

Codecheck Link: HERE and on Canvas

Participation Exercise 4-1b: `CircleAndSphere`

Goal: Write a program to display statistics for a circle and sphere of a given radius

```
Enter the radius: 5
The radius: 5.0
The circle circumference: 31.41592653589793
The circle area: 78.53981633974483
The sphere surface area: 314.1592653589793
The sphere volume: 523.5987755982989
```

Results of `CircleAndSphere.java`

The following formulas will be helpful:

Circle circumference: $2\pi r$

Circle area: πr^2

Sphere surface area: $4\pi r^2$

Sphere volume: $\frac{4}{3}\pi r^3$

Codecheck Link: HERE and on Canvas