CSE 121 – Lesson 8

Elba Garza & Matt Wang

Winter 2024


sli.do #cse121-8

Today’s playlist:
CSE 121 24wi lecture beats :D
Announcements, Reminders

• Alt Text video walk-through now available
• Programming Assignment 1 (P1) was due last night!
• Creative Project 2 (C2) releasing later today (due Feb 6th)
• Resubmission Cycle 1 (R1) due tomorrow, Feb 1st by 11:59pm
• Resubmission Cycle 2 (R2) opens tomorrow (due Feb 8th)
  • Eligible assignments: C0, P0, C1, P1
• Wednesday Feb 7th: Mid-term Formative Feedback in class
• Quiz 0 is tomorrow in your quiz section!
  • Quiz logistics announced on Ed
Common Problem-Solving Strategies

• **Analogy** – Is this similar to another problem you've seen?

• **Brainstorming** – Consider steps to solve problem before jumping into code
  • Try to do an example "by hand" → outline steps

• **Solve sub-problems** – Is there a smaller part of the problem to solve?

• **Debugging** – Does your solution behave correctly?
  • What is it doing?
  • What do you expect it to do?
  • What area of your code controls that part of the output?

• **Iterative Development** – Can we start by solving a different problem that is easier?
(Recall) Methods & Parameters

Definition: A value passed to a method by its caller

```java
public static void myMethod(String musicalAct) {
    System.out.print(musicalAct + " is the best!");
    ...
}
```

Calling a method with a parameter...

```java
myMethod("Olivia Rodrigo"); // Prints out
// "Olivia Rodrigo is
// the best!"
```
(Recall) Returns 1

Returns allow us to send values out of a method

```java
public static <type> myMethod(<zero or more params>) {
    ...
    return <value of correct type>
}
```

Calling a method that returns a value...

```java
$type result = myMethod(...);
```
(Recall) Returns 2

Returns allow us to send values out of a method

```java
public static String myMethod(String musicalAct) {
    System.out.print(musicalAct + " is the best!");
    ...
    return musicalAct + " is the best!"
}
```

Calling a method with a parameter...

```java
String s = myMethod("Olivia Rodrigo"); // Prints and returns // "Olivia Rodrigo is // the best!"
```
(Recall) Returns 3

Returns allow us to send values out of a method

```java
public static String myMethod(String musicalAct) {
    ...
    return musicalAct + " is the best!"
}
```

Calling a method with a parameter...

```java
String s = myMethod("Olivia Rodrigo"); // Returns
    // "Olivia Rodrigo is
    // the best!"
```
(Recall) Tricky Poll 1: Last line printed?

```java
public static final int COUNT = 7;
public static void main(String[] args) {
    int count = 5;
    line(count);
    System.out.println("count is: " + count);
}

public static void line(int count) {
    for (int i = 1; i <= count; i++) {
        System.out.print("*");
    }
    count++;
    System.out.println();
}
```
public static final int COUNT = 7;
public static void main(String[] args) {
    int count = 5;
    count = line(count);
    System.out.println("count is: " + count);
}

public static int line(int count) {
    for (int i = 1; i <= count; i++) {
        System.out.print("*");
    }
    count++;
    System.out.println();
    return count;
}
Tricky Poll 2: x’s, and y’s, and z’s, oh my.

```java
public static void main(String[] args) {
    int x = 9;
    int y = 2;
    int z = 5;

    mystery(z, y, x);
    mystery(y, x, z);
}

public static void mystery(int x, int z, int y) {
    System.out.println(z + " and " + (y - x));
}
```

```plaintext
9 2 5
5 9 2
```
Example of returns: Math class

<table>
<thead>
<tr>
<th>Methods</th>
<th>Returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math.abs(value)</td>
<td>Absolute value of value</td>
</tr>
<tr>
<td>Math.ceil(value)</td>
<td>value rounded up</td>
</tr>
<tr>
<td>Math.floor(value)</td>
<td>value rounded down</td>
</tr>
<tr>
<td>Math.max(value1, value2)</td>
<td>Larger of the two given values</td>
</tr>
<tr>
<td>Math.min(value1, value2)</td>
<td>Smaller of the two given values</td>
</tr>
<tr>
<td>Math.round(value)</td>
<td>value rounded to the nearest whole number</td>
</tr>
<tr>
<td>Math.sqrt(value)</td>
<td>Square root of value</td>
</tr>
<tr>
<td>Math.pow(base, exp)</td>
<td>base to the exp power</td>
</tr>
</tbody>
</table>
Math example

double value = 823.577564893;
double roundedValue = (double) Math.round(value * 100) / 100;
Poll in with your answer!

What is the correct implementation of a `maxDatingAge` method?

A. ```java
   public static void maxDatingAge(int age) {
     int maxDatingAge = age - 7 * 2;
     return maxDatingAge;
   }
```  

B. ```java
   public static void maxDatingAge(int age) {
     int maxDatingAge = age - 7 * 2;
   }
```  

C. ```java
   public static int maxDatingAge(int age) {
     int maxDatingAge = (age - 7) * 2;
     return maxDatingAge;
   }
```  

D. ```java
   public static int maxDatingAge(int age) {
     return (age - 7) * 2;
   }
```