

LEC 08

CSE 121

Methods, Parameters, and Returns

Questions during Class?

Raise hand or send here

sli.do **#cse121**



BEFORE WE START

Talk in chat:

What's your favourite book?

Music: 🌸 [CSE 121 25sp Lecture Tunes](#) 🌸

Instructor: Miya Natsuhara

TAs:	Chloë	Hibbah	Sushma
	Ailsa	Julia	Kelsey
	Johnathan	Sahej	Shayna
	Christian	Ruslana	Hannah
	Merav	Hanna	Zach
	Judy	Maitreyi	
	Janvi	Ayesha	

Agenda

- **Announcements, Reminders** ←
- Methods & Parameters Warmup
- Returns Review
- Temperatures Example
- Revisiting String Manipulation

Announcements, Reminders

- C2 (Social Network) releasing later today, due **Thursday** May 8th
 - C2 & P2 will have off-cycle release and/or due dates
 - for C2, expecting you *probably* won't start today; involves conditionals (Friday's topic)
- R1 due tomorrow; R2 opens tomorrow, due Thursday, May 8th
- Friday: guest lecture with Hannah again!
- Next Wednesday May 7th: in-class mid-quarter formative feedback
- Quiz 1: Thursday, May 15th

Agenda

- Announcements, Reminders
- **Methods & Parameters Warmup** ←
- Returns Review
- Temperatures Example
- Revisiting String Manipulation



Practice: Think

sli.do

#cse121

What is the output of this program?

```
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));  
}
```

A. 2 and 4
9 and 3

B. 5 and -7
5 and -7

C. 9 and -3
5 and -7

D. I'm lost



Practice: Pair

sli.do

#cse121

What is the output of this program?

```
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));  
}
```

A. 2 and 4
9 and 3

B. 5 and -7
5 and -7

C. 9 and -3
5 and -7

D. I'm lost

Agenda

- Announcements, Reminders
- Methods & Parameters Warmup
- **Returns Review** ←
- Temperatures Example
- Revisiting String Manipulation

PCM: Returns

Returns allow us to send values *out of a method*

```
public static <type> myMethod(...) {  
    ...  
    return <value of correct type>  
}
```

Evaluates the expression

Returns this value to where
the method is called from

Method immediately exits

Calling a method that returns a value...

```
<type> result = myMethod(...);
```

Recall: Math

Method	Description
<code>Math.abs(<i>value</i>)</code>	Returns the absolute value of <i>value</i>
<code>Math.ceil(<i>value</i>)</code>	Returns <i>value</i> rounded up
<code>Math.floor(<i>value</i>)</code>	Returns <i>value</i> rounded down
<code>Math.max(<i>value1</i>, <i>value2</i>)</code>	Returns the larger of the two values
<code>Math.min(<i>value1</i>, <i>value2</i>)</code>	Returns the smaller of the two values
<code>Math.round(<i>value</i>)</code>	Returns <i>value</i> rounded to the nearest whole number* note: need to cast result to int (it's complicated!)
<code>Math.sqrt(<i>value</i>)</code>	Returns the square root of <i>value</i>
<code>Math.pow(<i>base</i>, <i>exp</i>)</code>	Returns <i>base</i> raised to the <i>exp</i> power

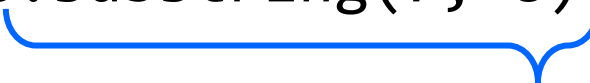
Recall: String Methods

Method	Description
<code>length()</code>	Returns the length of the string.
<code>charAt(<i>i</i>)</code>	Returns the character at index <i>i</i> of the string
<code>indexOf(<i>s</i>)</code>	Returns the index of the first occurrence of <i>s</i> in the string; returns -1 if <i>s</i> doesn't appear in the string
<code>substring(<i>i</i>, <i>j</i>)</code> or <code>substring(<i>i</i>)</code>	Returns the characters in this string from <i>i</i> (inclusive) to <i>j</i> (exclusive); if <i>j</i> is omitted, goes until the end of the string
<code>contains(<i>s</i>)</code>	Returns whether or not the string contains <i>s</i>
<code>equals(<i>s</i>)</code>	Returns whether or not the string is equal to <i>s</i> (case-sensitive)
<code>equalsIgnoreCase(<i>s</i>)</code>	Returns whether or not the string is equal to <i>s</i> ignoring case
<code>toUpperCase()</code>	Returns an uppercase version of the string
<code>toLowerCase()</code>	Returns a lowercase version of the string

Reminder: Gumball & Strings

```
String s = "bubblegum";
```

```
s = s.substring(7, 8).toUpperCase() + s.substring(8) + "ball";
```



```
s = "g".toUpperCase() + s.substring(8) + "ball";
```



```
s = "G" + s.substring(8) + "ball";
```



```
s = "G" + "um" + "ball";
```

Agenda

- Announcements, Reminders
- Methods & Parameters Warmup
- Returns Review
- **Temperatures Example** ←
- Revisiting String Manipulation

New: Method Comments

Each method you write (except main) should have a short comment!

```
// Behavior:  
// - Calculates net profit using monthly income and daily spending.  
// Parameters:  
// - income: user's income this month (non-negative)  
// - spending: amount spent each day this month (non-negative)  
// Returns:  
// - int: the net profit or loss. Positive if profit, negative if loss.  
public static int calculateNetExpenses(int income, int spending) {  
    return income - (spending * DAYS_IN_MONTH);  
}
```



Practice: Think

[sli.do](#)[#cse121](#)

What value is returned from this method?

```
public static int returnExample() {  
    for (int i = 0; i < 5; i++) {  
        return i;  
    }  
    return -1;  
}
```

A. -1

B. 0

C. 4

D. 5



Practice: Pair

sli.do

#cse121

What value is returned from this method?

```
public static int returnExample() {  
    for (int i = 0; i < 5; i++) {  
        return i;  
    }  
    return -1;  
}
```

A. -1

B. 0

C. 4

D. 5

Agenda

- Announcements, Reminders
- Methods & Parameters Warmup
- Returns Review
- Temperatures Example
- **Revisiting String Manipulation** ←

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?