

CSC207.01 2013F, Class 10: Exceptional Programming

Overview

- Preliminaries
 - Admin.
 - Questions on HW3
- Dealing with errors.
- Java's Exceptions.
- Lab.

Admin

- New partners for this week's labs.
- Dutch letters! Take half of one (if you'd like).
- Study break today at 8pm.
- Picnic Friday!
- VNC Instructions at <http://www.cs.grinnell.edu/vnc>
- Exception handling is one of the first places where my students start saying "Java is your nanny." Can you tell why?
- I managed to finish reviewing HW1 this weekend. You should have my responses. You can also see answers to most questions in the FAQ.
- I'll reserve time in class today for questions on HW3.
- I expect to have the next homework available tomorrow.
- Readings for Tuesday:
 - Writing Your Own Classes
 - Standard Object Methods
- EC Opportunities
 - Convocation this Wednesday at noon. (Sam rants about the importance of convo. See the outline for details.)
 - Learning from Alumni Thursday @ 2:15 Sam Tape and company.
 - CS Extras Thursday @ 4:30: Kim Spasaro on Linguistics Programming
 - CS Table Friday (pair programming)
 - Participate in Humans vs. Zombies. And yes, you can still get in.
 - Other?

Questions on HW3

- What should we return for part (the maknig change question): Choose one of these:
 - An array of numbers that give the count of each coin. 2, 7, 11, 54
 - For 28: 0, 4, 0, 0
 - Or 0, 28, 0, 0

- An array of numbers that give the list of coin values.
- DOCUMENT!
- Should we cite that Java API? It depends on how you use it. If you just look something up, I wouldn't bother. If you copy code, I would cite it.

Dealing with errors

- Some errors are the programmer's fault. We call those bugs.
- Some errors are beyond the programmer's control
 - Bad inputs
 - Missing files
- We need to think about errors
- Sam's model: Your code gets used by other code, gets used by the GUI, interacts with user
 - YOUR UTILITY CODE SHOULD NOT USE PRINTF TO REPORT ERRORS!
- In Java, we have Exceptions/Errors to report errors
 - Caller provides two places to return: Success/Failure

A procedure that can fail

```
public static type proc(inputs) throws Exception
```

Fail

```
throw new Exception("reason");
```

Dealing with errors - throw your own

```
public static type caller(inputs) throws Exception { proc(...); }
```

Dealing with errors - Provide the alternate flow of control

```
public static type caller(inputs) { try { proc(...); ...; } catch (Exception e) { // RECOVER } }
```

Lab

- Question 1: Autoflush makes it print output as soon as you print a newline. (or at least use println).
 - If you don't flush or close at the end, your last output gets dropped.

Copyright (c) 2013 Samuel A. Rebelsky.



This work is licensed under a Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.