

A COMPETITIVE-COLLABORATIVE APPROACH FOR INTRODUCING SOFTWARE ENGINEERING IN A CS2 CLASS

Swapneel Sheth, Jonathan Bell, Gail Kaiser
Department of Computer Science
Columbia University
{swapneel, jbell, kaiser}@cs.columbia.edu

@swapneel, @_jon_bell_

MOTIVATION

- Students at universities are increasingly competitive
- Grades are typically a zero-sum game
- This competition can discourage communication among students
- Competition may not lead to better learning

APPROACH

- Use competition to help students learn material better
- Encourage more communication by introducing collaborative aspects

COMS 1007

- Object Oriented Programming and Design In Java
- Second course for CS Majors and Minors at Columbia University (required for other engineering majors; also fulfills general science, CS requirement for non-engineering)
- Taught in Spring 2012
- Enrollment 129 (largely freshmen and sophomores)

COMS 1007

- Course Material
 - Object Oriented Design, Design Patterns, Interfaces, Graphics Programming, Inheritance and Abstract Classes, Networking, and Multithreading and Synchronization
- 5 biweekly assignments (theory and programming)
- In-class midterm and final

COMPETITION

- Battleship Tournament (Software Design, Interfaces)
- Gamification using HALO (Software Testing)

ANCHOR
MB
KEY
TO FUN
and learning

FOR
AGES
8 TO
ADULT

MILTON
BRADY
COMPANY
SPRINGFIELD
MASSACHUSETTS
4730
MADE IN U.S.A.

AMERICA'S ALL TIME
FAVORITE GAME

BATTLESHIP

IT'S A HIT

G-4



For
2
players

OBJECT
Sink all of your
opponent's ships

average playing time 20 minutes

MB
MILTON
BRADY
COMPANY

BATTLESHIP

GAME



LEARNING GOALS

- Good Design
- Using Interfaces
- “Program to an interface, not an implementation”


```

1 /**
2  * The game interface – this will control the Battleship
   game.
3  * It will keep track of 2 versions of the "board" – one
   for each player.
4  * It will let players take turns.
5  * It will announce hits , misses , and ships sunk (by
   calling the appropriate methods in the Player
   interface/class).
6  * @author swapneel
7  *
8  */
9 public interface Game {
10
11     int SIZE = 10;
12
13     int CARRIER = 5;
14     int BATTLESHIP = 4;
15     int SUBMARINE = 3;
16     int CRUISER = 3;
17     int DESTROYER = 2;
18
19     /**
20      * This method will initialize the game.
21      * At the end of this method, the board has been
   set up and the game can be started
22      * @param p1 Player 1
23      * @param p2 Player 2
24      */
25     void initialize (Player p1, Player p2);
26
27     /**
28      * This is the start point of playing the game.
29      * The game will alternate between the players
   letting them take shots at the other team.
30      * @return Player who won
31      */
32     Player playGame ();
33
34 }

```



```

1 /**
2  * The Location interface to specify how x and y
3  * coordinates are represented.
4  * This can be used to represent the location of a ship
5  * or a shot.
6  * If the location is a shot, the isShipHorizontal()
7  * method can return an arbitrary value.
8  * @author swapneel
9  *
10 */
11 public interface Location {
12
13     /**
14      * Gets the x coordinate
15      * @return the x coordinate
16      */
17     int getX();
18
19     /**
20      * Gets the y coordinate
21      * @return the y coordinate
22      */
23     int getY();
24
25     /**
26      * This method will indicate whether the ship is
27      * horizontal or vertical.
28      * Can return an arbitrary value if the location
29      * is used to indicate a shot (and not a ship)
30      * @return true if ship is horizontal, false
31      * otherwise
32      */
33     boolean isShipHorizontal();
34 }

```



```

1 /**
2  * The Player interface
3  * Each player will get to choose where to place the 5
4     ships and how to take turns shooting at enemy ships
5  * @author swapneel
6  */
7 public interface Player {
8
9     /**
10    * This method will place a ship on the grid.
11    * This method should guarantee correctness of
12       location (no overlaps, no ships over the
13       edge of the board, etc.)
14    * @param size the size of the ship to place
15    * @param retry if an earlier call to this
16       method returned an invalid position, this
17       method will be called again with retry set
18       to true.
19    * @return The Location of the ship
20    */
21    Location placeShip(int size, boolean retry);
22
23    /**
24    * This method will get the new target to aim for
25    * @return The Location of the target
26    */
27    Location getTarget();
28
29    /**
30    * This method will notify the Player of the
31       result of the previous shot
32    * @param hit true, if it was a hit; false
33       otherwise
34    * @param sunk true, if a ship is sunk; false
35       otherwise
36    */
37    void setResult(boolean hit, boolean sunk);
38 }

```

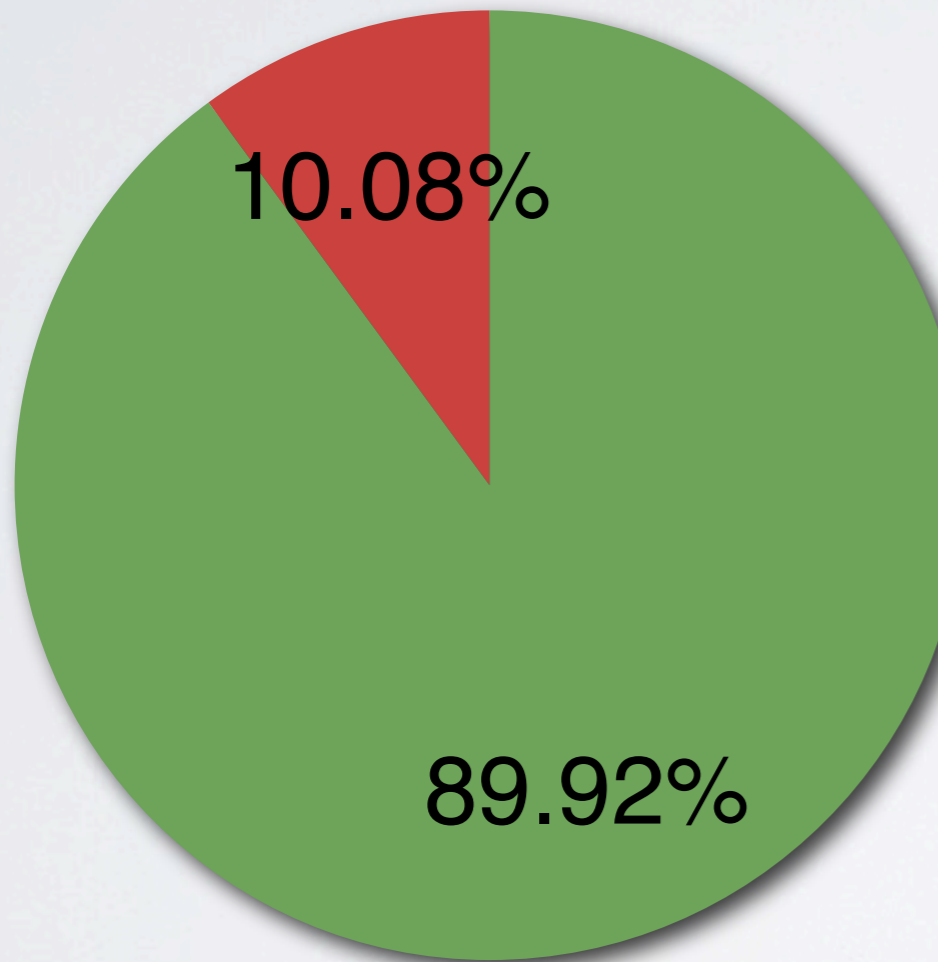
TOURNAMENT

- 1000 games against simple AI player written by TA
- Used as seeding for Single Elimination Bracket
- Players with good strategies would win, proceed to next round against other students
- Quarterfinals, Semifinals, Finals in class
- Extra Credit points based on results of the tournament

RESULTS

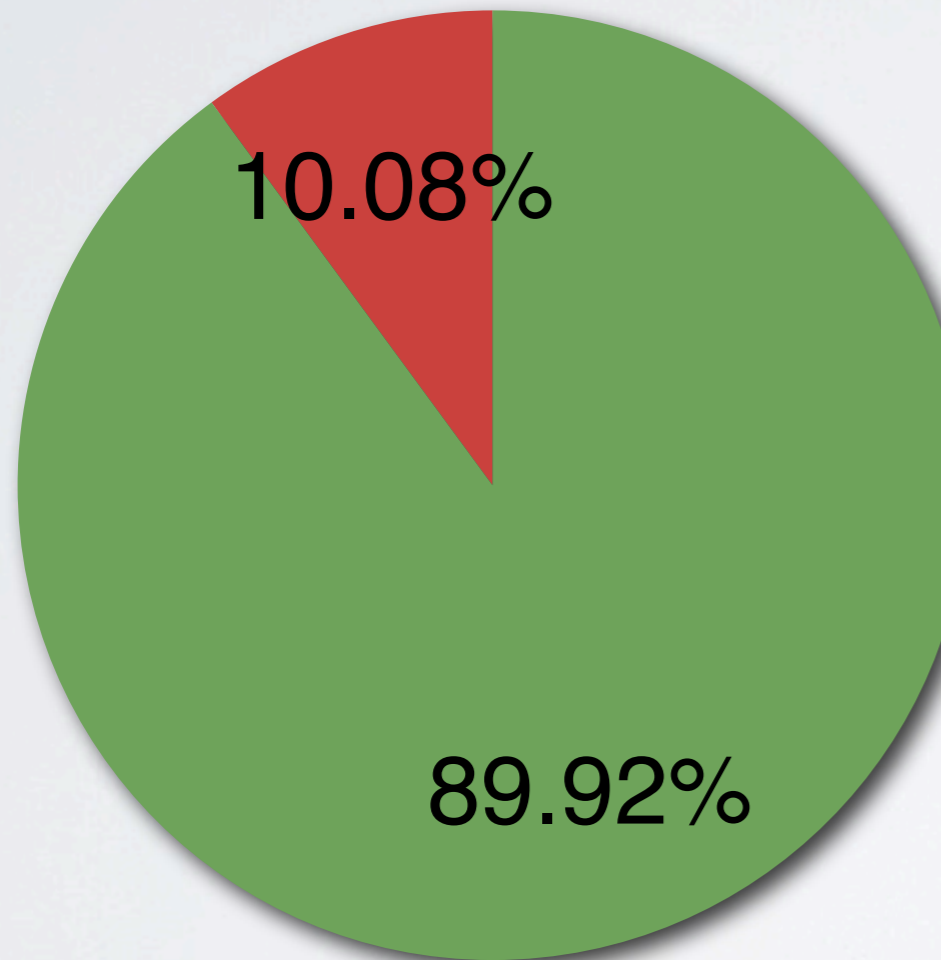
RESULTS

Wanted to take part in the tournament

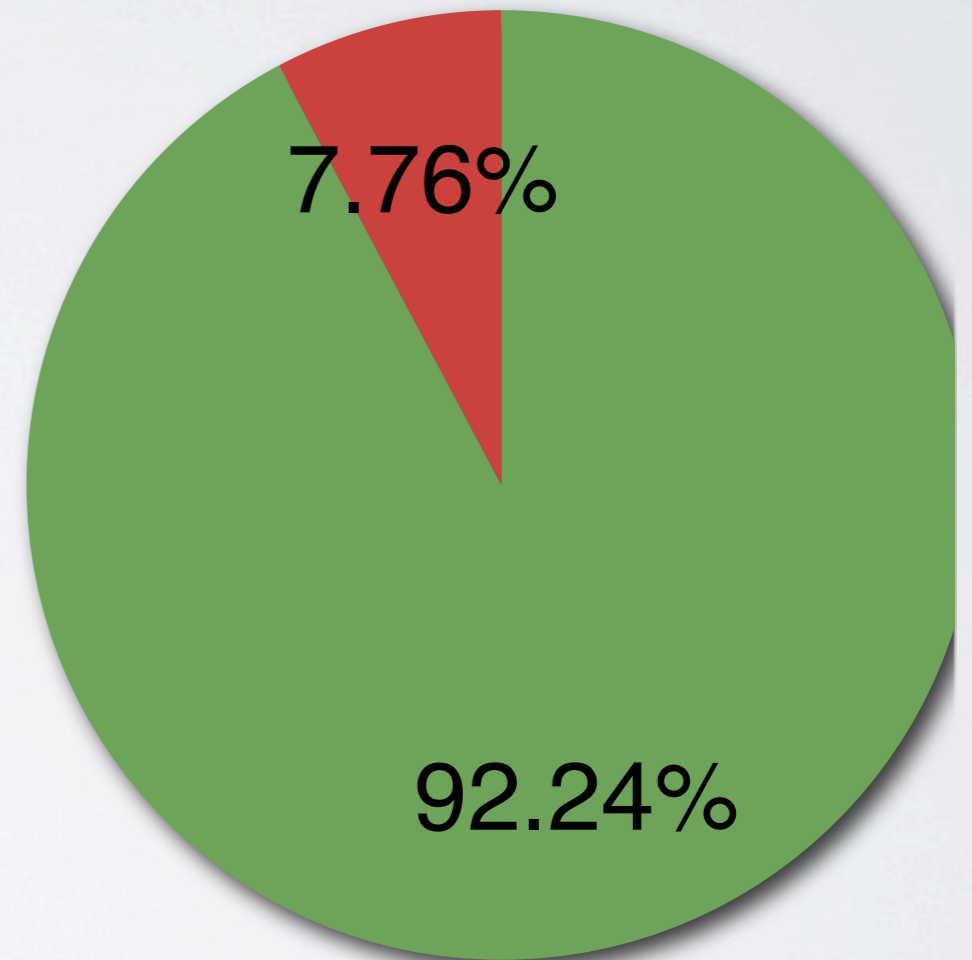


RESULTS

Wanted to take part in the tournament



Able to take part in the tournament



GAMIFICATION USING HALO

- Introductory CS classes do not focus on software testing [Jackson et al.; Elbaum et al.; Jones]
- “If it compiles and run without crashing, it must work fine”
- Low student interest and engagement in software testing [Elbaum et al.]

HALO SOFTWARE ENGINEERING

- Highly Addictive socialLly Optimized Software Engineering
- Use game-like elements and motifs to make software testing more fun, social, and engaging
- Quests disguise testing techniques like boundary value analysis
- Students complete quests => code gets tested better

HALO IN COMS 1007

- Halo Quests optional for HW2 and HW3
- For HW4, a small amount of Extra Credit could be earned
- For HW5 (Team Projects), students had to create HALO quests for Extra Credit
- “Learning by example”

Java - invivo-expreval/src/edu/columbia/psl/invivoexpreval/samples/InVivoExprEvalExample.java - Eclipse - /Users/jon/Documents/workspace

Package Explorer: cloning, columbus2-runtime, columbus2-tester, commons-compiler, src, edu.columbia.psl.commons.compiler, AbstractCompilerFactory.java, AbstractJavaSourceClassLoader.java, CausedException.java, CompileException.java, CompilerFactoryFactory.java, Cookable.java, IClassBodyEvaluator.java, ICompilerFactory.java, IKookable.java, IExpressionEvaluator.java, IScriptEvaluator.java, ISimpleCompiler.java, LocatedException.java, Location.java, PrimitiveWrapper.java, JRE System Library [Java SE 6 (MacOS X Default)], invivo-expreval, src, edu.columbia.psl.invivoexpreval, edu.columbia.psl.invivoexpreval.samples, InVivoExprEvalExample.java, package.html, edu.columbia.psl.invivoexpreval.util, edu.columbia.psl.invivoexpreval.util.enumerator, edu.columbia.psl.invivoexpreval.util.iterator, edu.columbia.psl.invivoexpreval.properties, Referenced Libraries, JRE System Library [Java SE 6 (MacOS X Default)], build, lib, invivo-junit-runner, invivo-junit-tester, invivo-runtime, invivo-runtime-c, jass.modern.core, mtnedew-runtime, mtnedew-tester, native-detector, tomcat

```

1
2
3
4 package edu.columbia.psl.invivoexpreval.samples;
5
6 import java.io.*;
7
8
9
10
11
12
13
14 public class InVivoExprEvalExample extends Traverser {
15     public static class foo {
16         public static void bar() {
17             System.out.println("Check");
18         }
19     }
20     public static void main(String[] args) throws CompileException, IOException {
21         ExprEvaluator ee = new ExprEvaluator(
22             "c > d ? c : d", // expression
23             int.class, // expressionType
24             new String[] { "c", "d" }, // parameterNames
25             new Class[] { int.class, int.class } // parameterTypes
26         );
27
28         Integer res = null;
29         try {
30             res = (Integer) ee.evaluate(
31                 new Object[] { // parameterValues
32                     new Integer(10),
33                     new Integer(11),
34                 }
35             );
36         } catch (InvocationTargetException e) {
37             // TODO Auto-generated catch block
38             e.printStackTrace();
39         }
40         System.out.println("res = " + res);
41
42         ee = new ExprEvaluator(
43             "a.bar()", // expression
44             void.class, // expressionType
45             new String[] { "a" }, // parameterNames
46             new Class[] { foo.class } // parameterTypes
47         );
48
49         try {
50             res = (Integer) ee.evaluate(
51                 new Object[] { // parameterValues
52                     new foo()
53                 }
54             );
55         }
56     }
57 }

```

Outline: Ant, HALO Quests, Bytecode Reference

- Assignment 2 (due 2 months, 1 week and 6 days ago)
- Assignment 3 (due 1 month, 1 week and 5 days ago)
- Assignment 4 (due 3 weeks and 2 days ago)
 - TARDIS
 - The Unicorn and the Wasp
 - New Earth
 - EXTERMINATE! EXTERMINATE!
 - Partners in Crime or your Companion
 - Blink
 - The Sonic Screwdriver
 - Human Nature
 - The Sontaran Stratagem
 - Amy's Choice

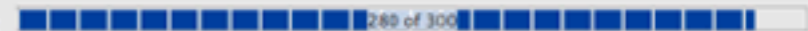
EXTERMINATE! EXTERMINATE!
(Part of Assignment 4, due 3 weeks and 2 days ago)


Background
The CIA factbook has some unstructured data - not all of it is organized properly. This may not be as annoying (or life threatening) as Daleks, but your programs should be able to deal with this correctly and not crash (or get exterminated).
Completing this quest will reward you with 30 XP

Objectives
Partners in Crime or your Companion
Blink

Problems, Javadoc, Declaration, Console, Plug-in Dependencies, Search, Debug, Bytecode, Servers, HALO Dashboard

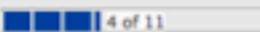
Jon Bell Level 3 Level 4




Experience Points:  280 of 300

Achievements  200 points of 330

Recent Achievements:

- The hardest part... nah, the easiest part! - May 15, 2012
- Conquered Square One - Feb 27, 2012
- The Tree Walker - Nov 11, 2011
- Time Cop - Nov 11, 2011

Quest Progress  4 of 11

- Assignment 2  3 of 5 quests Due 2 months, 1 week and 6 days ago
- Assignment 4  0 of 3 quests Due 3 weeks and 2 days ago
- Assignment 3  1 of 3 quests Due 1 month, 1 week and 5 days ago

Leaders
(By XP points; top 30)

- 1: Brian Wu (1085)
- 2: Kyle L Gelnett (945)
- 3: Angela Kay Clague (915)
- 4: Maxwell Hume (915)
- 5: Christopher Scott So (915)
- 6: Shensi Ding (905)
- 7: Pin-Joe Ko (905)
- 8: Vanshil Shah (905)
- 9: Daniel Kaplan (880)
- 10: Ruriko Araki (875)
- 11: Tanvi Bikhchandani (875)

CIA WORLD FACT BOOK

Publications Additional CIA Maps

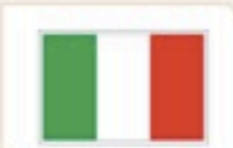
THE WORLD FACTBOOK

--- SELECT A COUNTRY OR LOCATION ---

ABOUT REFERENCES APPENDICES FAQs CONTACT VIEW TEXT/LOW BANDWIDTH VERSION
DOWNLOAD PUBLICATION


EUROPE :: ITALY PRINT

PAGE LAST UPDATED ON MAY 7, 2013




(CONTAINS DESCRIPTION)

CLICK FLAG OR MAP TO ENLARGE



CLICK FLAG OR MAP TO ENLARGE



CLICK MAP TO ENLARGE

EXPAND ALL | COLLAPSE ALL

- Introduction :: ITALY +
- Geography :: ITALY +
- People and Society :: ITALY +
- Government :: ITALY +
- Economy :: ITALY +
- Energy :: ITALY +
- Communications :: ITALY +
- Transportation :: ITALY +
- Military :: ITALY +
- Transnational Issues :: ITALY +

EXPAND ALL | COLLAPSE ALL

CIA WORLD FACT BOOK ASSIGNMENT

- Use Java Networking to fetch data
- Parse HTML
- Answer Jeopardy-style questions

CIA WORLD FACT BOOK ASSIGNMENT

- List countries in *South America* that are prone to *earthquakes*
- Find the country with the lowest elevation point in *Europe*
- A landlocked country is one that is entirely enclosed by land. For example, Austria is landlocked and shares its borders with Germany, Czech Republic, Hungary, etc. There are certain countries that are entirely landlocked by a single country. Find these countries.



HALO QUESTS

- **TARDIS** - java program that can Transfer And Read Data from Internet Sites
- New Earth — This will probably be your first program that talks to the Internet. While this isn't as complex as creating a new Earth, you should test out the basic functionality to make sure it works. Can you program read one page correctly? Can it read multiple pages? Can it read all of them?

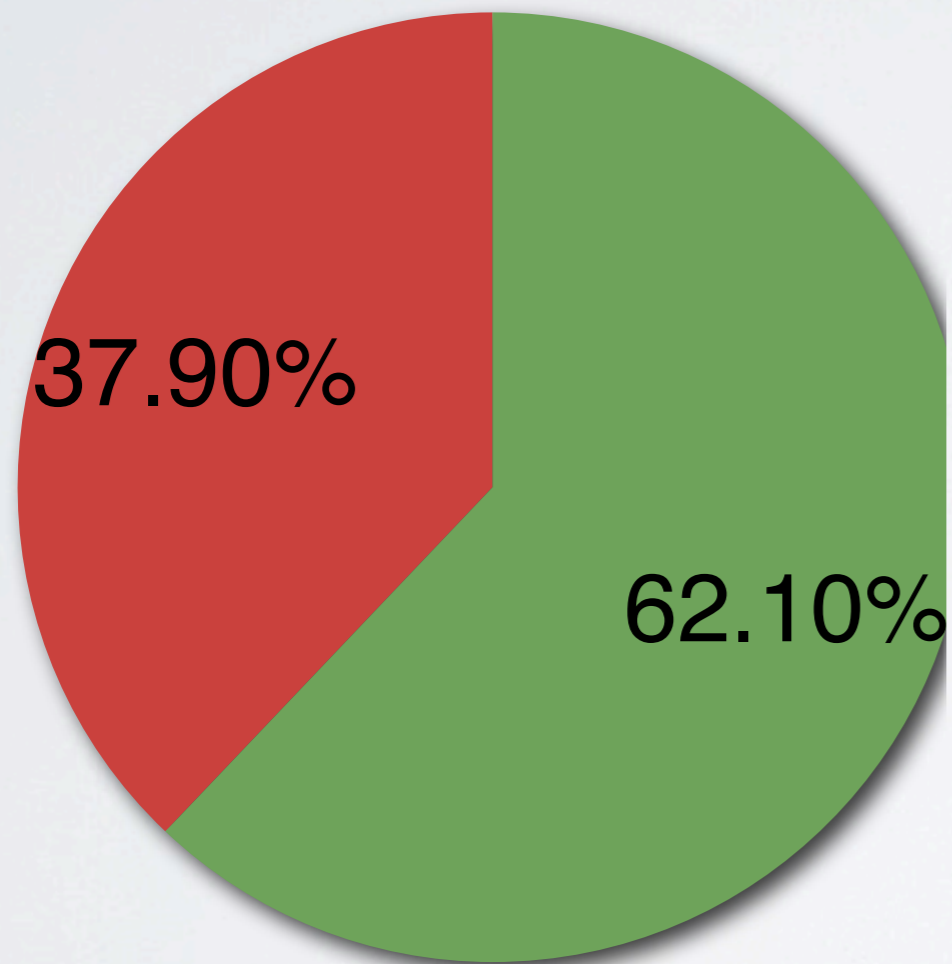
HALO QUESTS (2)

- **EXTERMINATE! EXTERMINATE!** - The CIA factbook has some unstructured data - not all of it is organized properly. This may not be as annoying (or life threatening) as Daleks, but your programs should be able to deal with this correctly and not crash (or get exterminated).
- Blink — Your program doesn't need to be afraid of the Angels and can blink, i.e., take longer than a few seconds to run and get all the information. However, this shouldn't be too long, say 1 hour. Does your program run in a reasonable amount of time?

RESULTS - HW 5

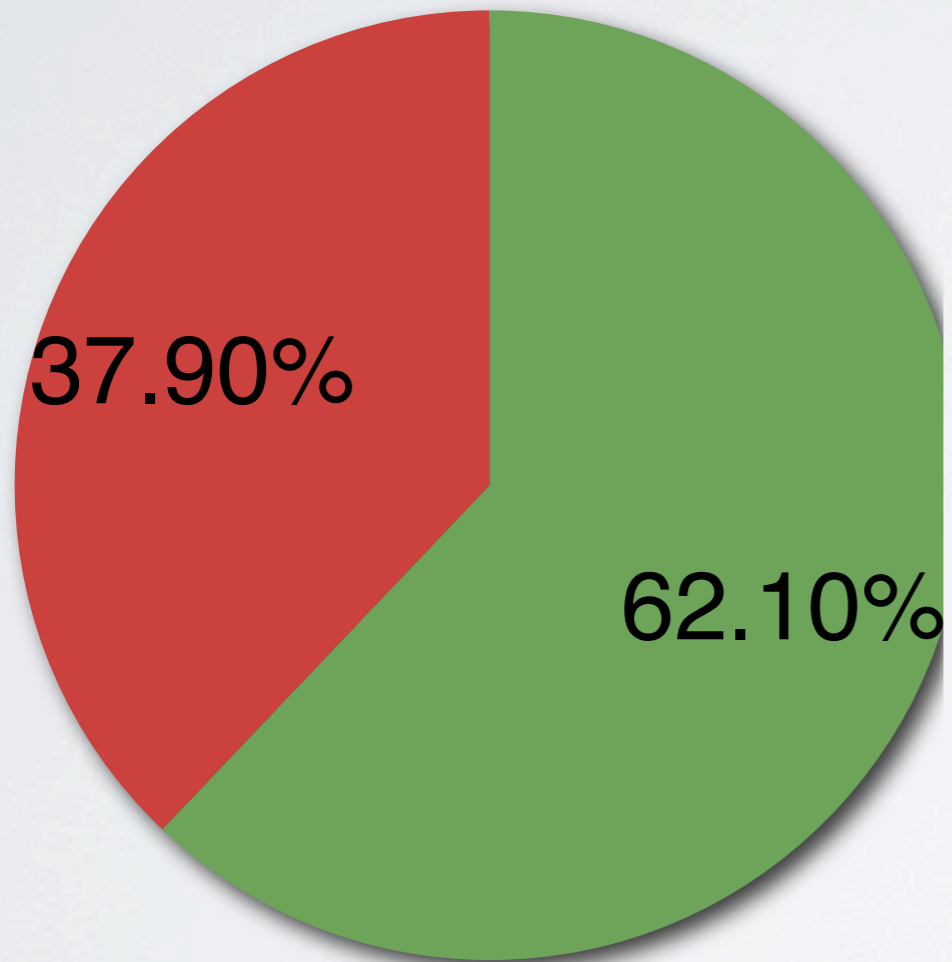
RESULTS - HW 5

Attempted the Extra Credit

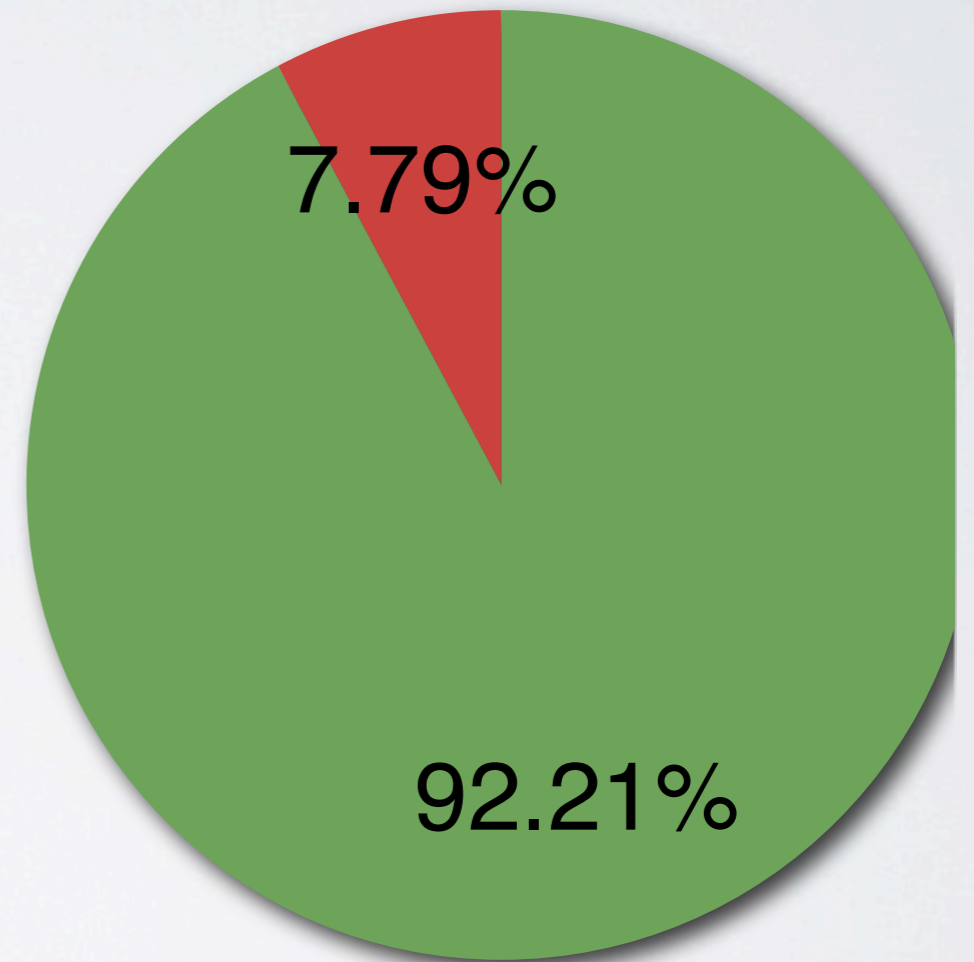


RESULTS - HW 5

Attempted the Extra Credit



Perfect Score on the Extra Credit



COLLABORATION

- Team Projects
- Lectures in Class

TEAM PROJECTS

- Most introductory CS courses allow little/no collaboration
- Real world projects are typically done in (large) teams
- Learn good (and bad) aspects about working in teams
- Added pedagogical incentive - give students the freedom to choose and deal with ambiguity that results

TEAM PROJECTS

Formal Design (CRC, Class diagrams, Sequence/State Diagrams)

Interfaces (Define and use)

Inheritance

Design Patterns

Java Graphics

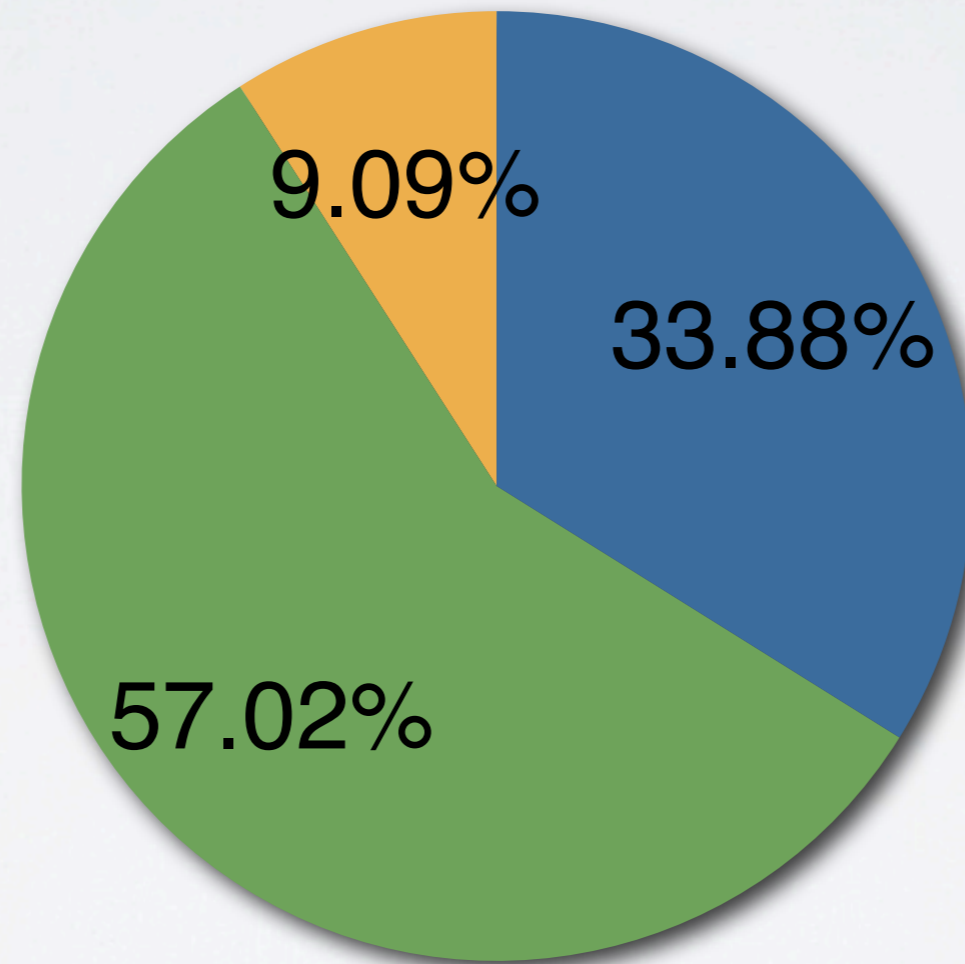
Networking

Multithreading

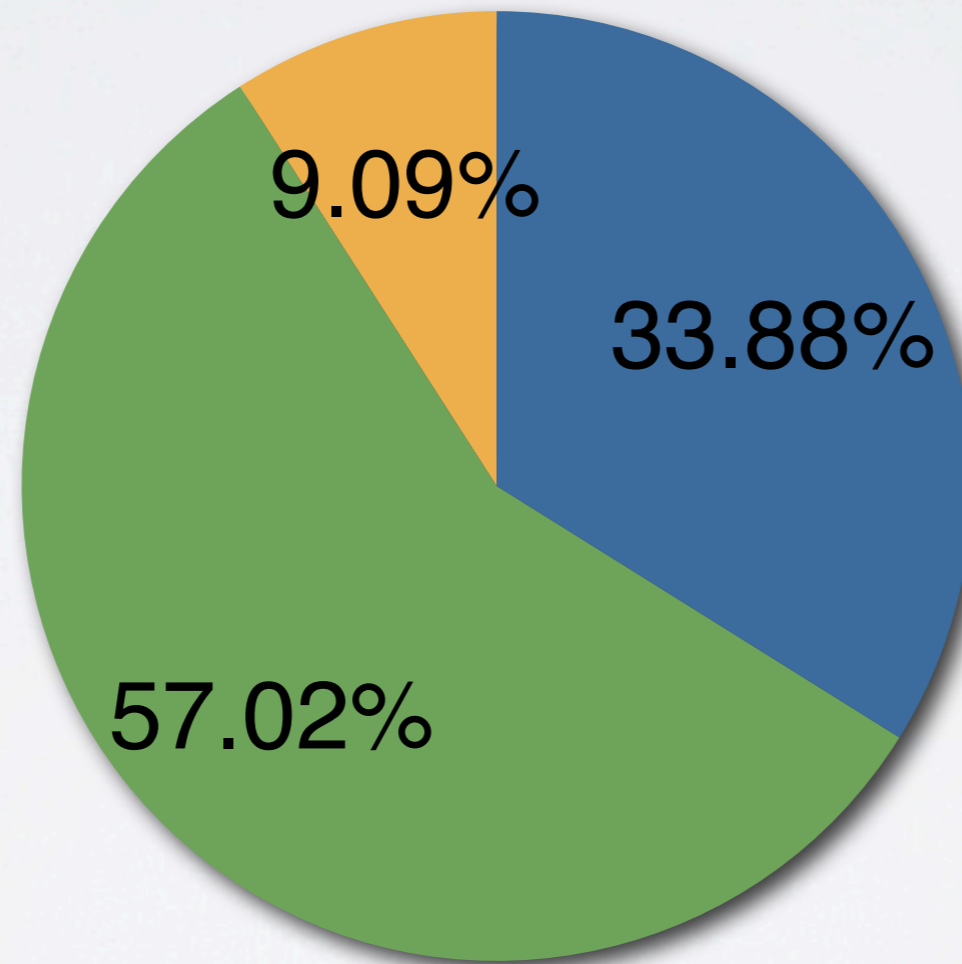
Advanced Java (Data Structures, Reflection, External Libraries, etc.)

TEAM PROJECTS

TEAM PROJECTS



TEAM PROJECTS



- Default Individual Assignment
- Custom Project, Team Size 2+
- Custom Project, Team Size 1

SAMPLE TEAM PROJECTS

- **Meal Planner** - use the USDA nutrition database, allow users to create and share recipes, learn about percentage values of food
- Navigate and read **SparkNotes** book summaries, allowing for offline use on a subway
- **What I should wear** - Clothes-recommendation system based on local weather
- Games like Guess Who?, Multiplayer Minesweeper, Mafia

LECTURES IN CLASS

- Two 75-minute lectures per week
- Stereotypes about CS being “geeky” and “boring” are not reinforced
 - Informal, collaborative classroom environment
 - Students encouraged to participate/comment
 - Group exercises

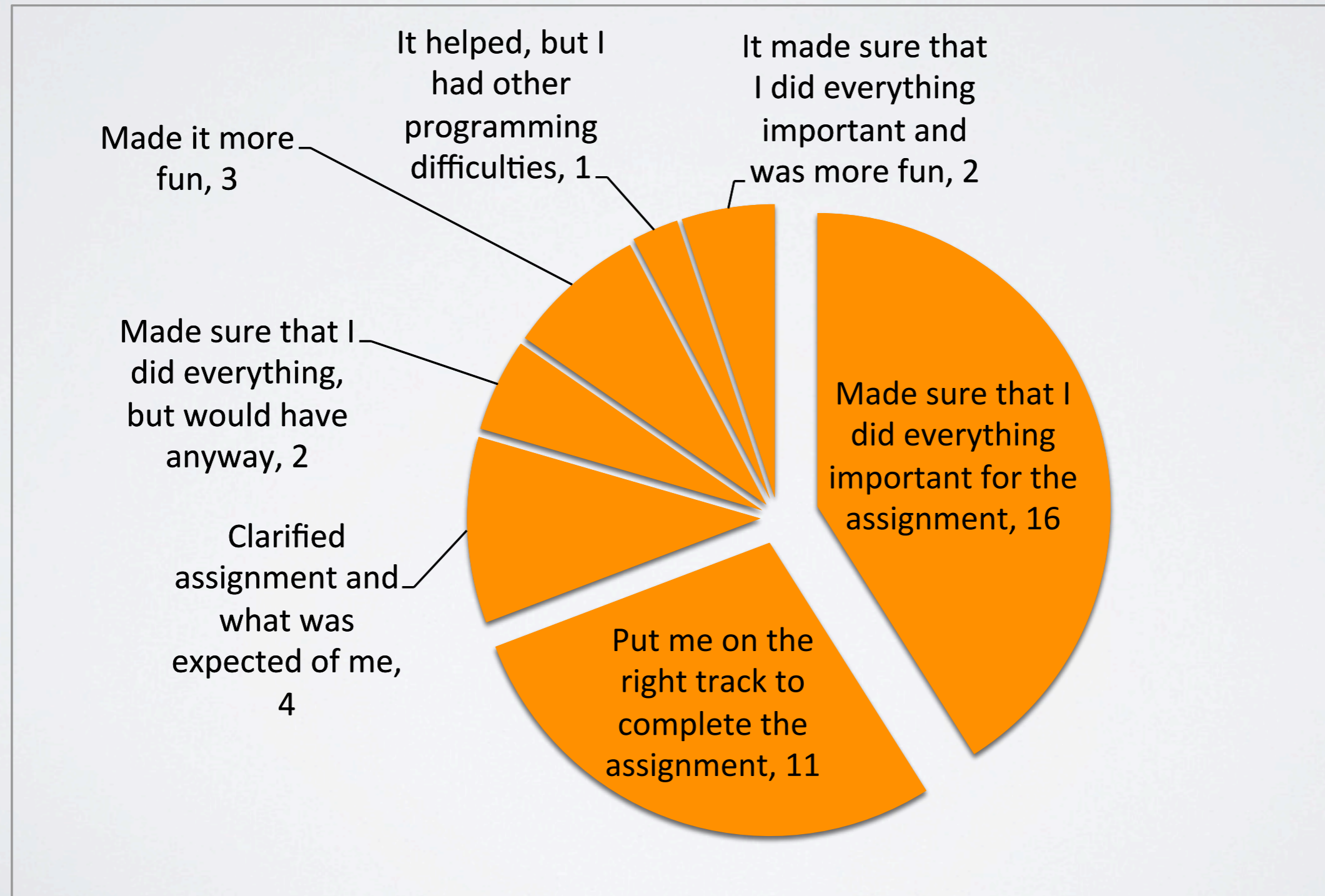
LECTURES IN CLASS

- Assignments are well-defined and precise, but real-world projects are typically not
 - All assignments had elements of vagueness
 - Team projects
 - Lectures reinforced the importance of being flexible and accepting uncertainty is part of CS

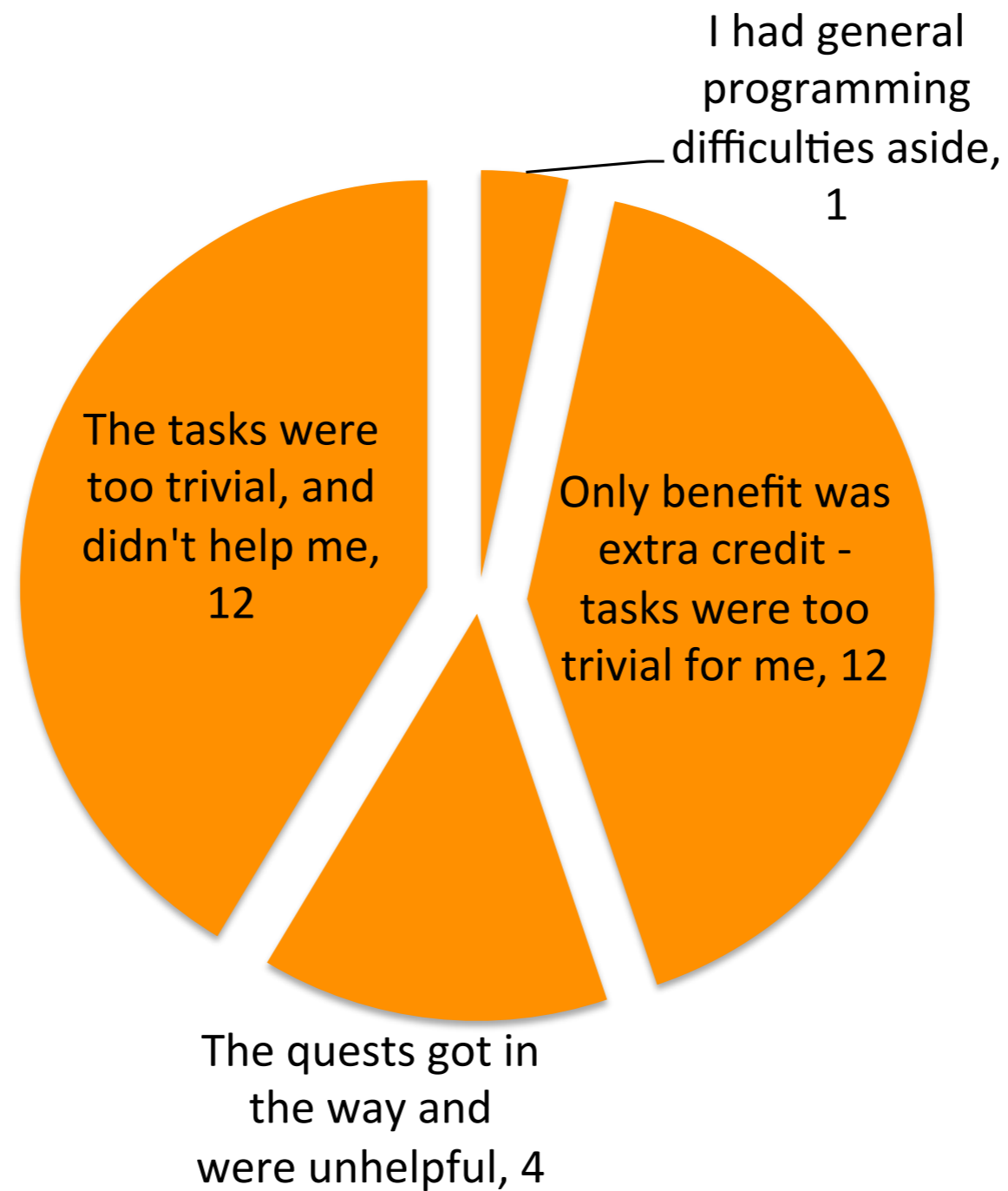
FEEDBACK - QUALITATIVE

- Sources
 - Midterm and final course evaluations
 - Surveys filled out by students
 - Email

FEEDBACK - HALO



FEEDBACK - HALO



FEEDBACK - COMPETITIVE ASPECTS

- “I really liked the class tournaments. If only there was a way to make them like mandatory.”
- “I think it’s awesome that you’re sneaking your taste in music into the HALO quests. The Coldplay references are hilarious. PLEASE make every HALO quest music-themed. It keeps me awake and happy as I do my homework.”

FEEDBACK - COLLABORATIVE ASPECTS

- “It’s really rare to feel like learning is a collaborative experience between the professor and the students, and even rarer to feel like it’s a collaborative experience amongst the students themselves, and you’ve been awesome at fostering that kind of environment.”

CONCLUSION

- Competition
 - Battleship Tournament (Software Design, Interfaces)
 - Gamification using HALO (Software Testing)
- Collaboration
 - Team Projects
 - Lectures in Class

enable (v_t) : to make possible, practical, or easy



PROGRAMMING SYSTEMS LAB
COLUMBIA UNIVERSITY



<http://www.psl.cs.columbia.edu/>

A COMPETITIVE-COLLABORATIVE APPROACH FOR INTRODUCING SOFTWARE ENGINEERING IN A CS2 CLASS

Swapneel Sheth, Jonathan Bell, Gail Kaiser
Department of Computer Science
Columbia University
{swapneel, jbell, kaiser}@cs.columbia.edu

@swapneel, @_jon_bell_