Discussion 6

Inheritance + Midterm Review

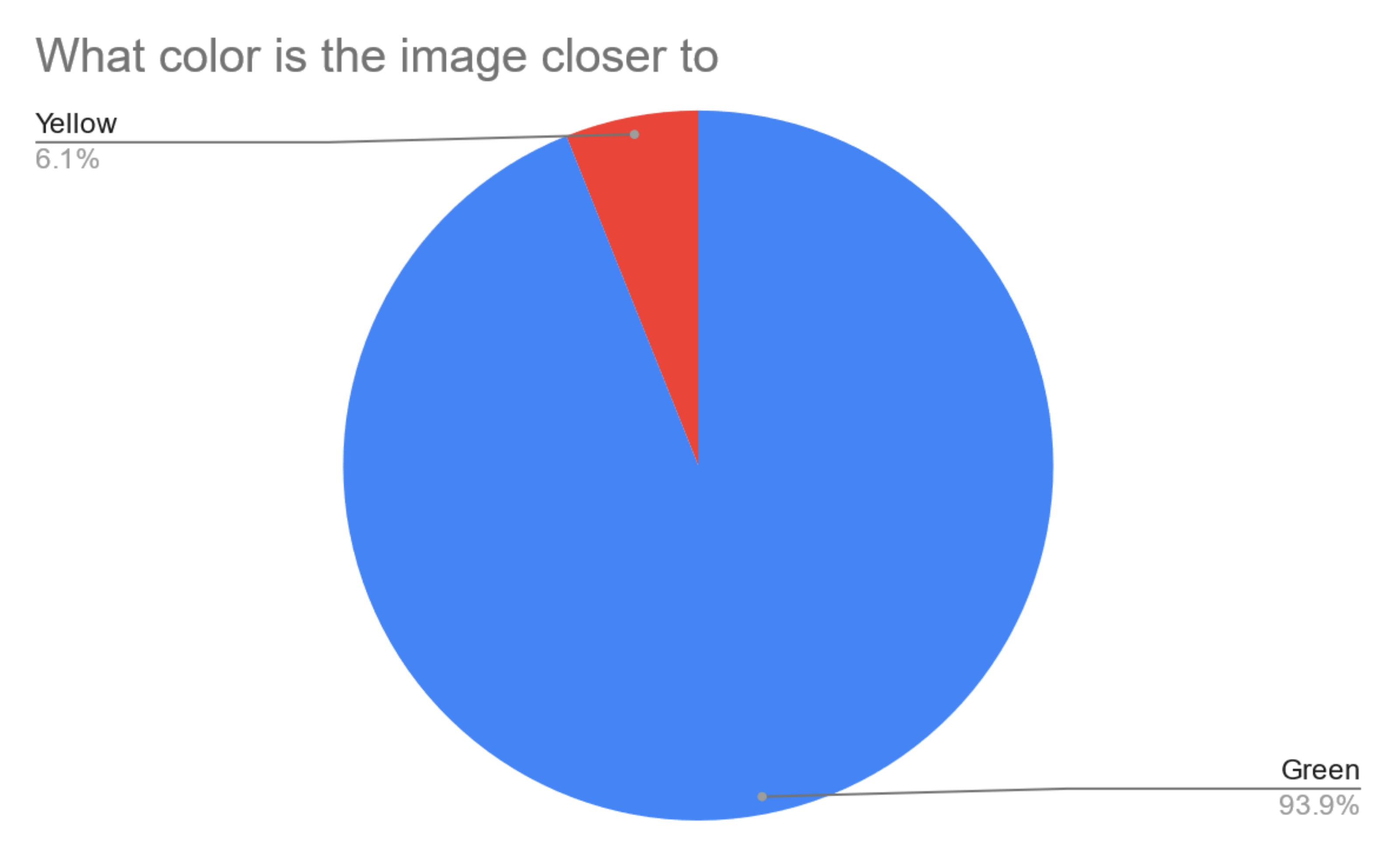
Antonio Kam

`anto [at] berkeley [dot] edu`

All slides can be found on

`teaching.rouxl.es`

Result from last discussion



My results from North American Championships!

- 3x3 One-Handed
    - Didn't break any personal records 😭
    - Ended up coming 4th 💀 (would have been on the podium if I got a result 0.1s faster than what I got in finals)
    - I'm now the fastest loser! (Fastest ever 4th place finish 👀)
- 3x3 Blindfolded
    - Broke the Asian Record for single time (17.47s)
    - Came 3rd in the final round 🤯 (this was very unexpected because I thought I'd place around 6-8 or so)
    - Part of the fastest podium ever!

Questions and Comments from last discussion

- Inheritance/OOP is a difficult concept
  - I'll show you something that I like to do when it comes to object diagrams that I think is very helpful
- Tree Recursion/Recursion
- Lists + Environment Diagrams
  - Might do this with WWPD questions from past midterms, or something of a similar difficulty
- HOFs
  - Will see if there's enough time for this

Announcements

- Midterm on Thursday! Good luck 🐈
- No discussion on Thursday
- Lab 04 due today 🥶
- Lab 05 and HW 03 due **Wednesday**
  - Not Thursday!!!
- This discussion and next lab will be targeted towards Midterm Review
- There will be 15 minutes of time at the end of this discussion to fill in the mid-semester feedback form + Go over my study tips 👀
  - Will be super useful in improving your experience with CS 61A

Temperature Check 🌡️

😄 Let's see what you all want to do

- Dictionaries? Yes/No
  - Keep in mind time constraints!

Inheritance

```python
class Dog():
    def __init__(self, name, owner):
        self.is_alive = True
        self.name = name
        self.owner = owner
    def eat(self, thing):
        print(self.name + " ate a " + str(thing) + "!")
    def talk(self):
        print(self.name + " says woof!")

class Cat():
    def __init__(self, name, owner, lives=9):
        self.is_alive = True
        self.name = name
        self.owner = owner
        self.lives = lives
    def eat(self, thing):
        print(self.name + " ate a " + str(thing) + "!")
    def talk(self):
        print(self.name + " says meow!")
```

Inheritance

Notice the redundancies in the code? One of the core foundations in this class is to not repeat yourself (DRY)

- Instead, you can use inheritance to solve this problem
- Syntax when creating a class is to put brackets around the class you want to inherit:

```
class Cat(Animal): # Cat inherits the Animal class - as in, all cats are animals
    ...
```

Inheritance

```python
class Pet():
    def __init__(self, name, owner):
        self.is_alive = True    # It's alive!!!
        self.name = name
        self.owner = owner
    def eat(self, thing):
        print(self.name + " ate a " + str(thing) + "!")
    def talk(self):
        print(self.name)

class Dog(Pet): # Inherits all methods/variables from the Animal class
    def talk(self):
        print(self.name + ' says woof!')
```

Inheritance - `super()`

- Calling `super()` will refer to the class's *superclass*
- You can use the parent's method and then add on to that.

```
class Class(Pet): # Inherits all methods/variables from the Animal class
    def __init__(self, name, owner, lives = 9):
        super().__init__(name, owner)
        # same as calling Pet.__init__(self, name, owner) from here
        self.lives = 9
    def talk(self):
        print(self.name + ' says meow!')
```

Worksheet!

Attendance

`links.rouxl.es/disc`

Worksheet!

Mid-semester survey

`go.cs61a.org/mid-sem-survey`

Midterm Tips

- Take a practice midterm as a diagnostic
- Grade it honestly (better to know what you don't know now rather than during the midterm)
- Understand the solutions for those problems (the videos/guides might be very useful), and find other similar problems
- Do another practice midterm and see if you improved
- Do the questions that you missed before (without the solutions) and see if you understand it better on your second pass

Midterm Tips

- Please remember to sleep well before the midterm! It makes a serious difference
- Don't touch CS for a few hours before the midterm - it won't be as effective, and you won't learn too much
- I tend to like putting midterm problems that I thought were interesting/challenging on my cheat sheet

Midterm Tips

- Please use pencil, especially for environment diagrams!
- Don't leave any line empty - there's a lot of partial credit
- Read the problem description carefully, and look at the doctests in detail
- Pay attention to data types/inputs/outputs

Mental Health Resources

- CAPS:
  - If you need to talk to a professional, please call CAPS at 510-642-9494.
- After Hours Assistance
  - For any assistance after hours, details on what to do can be found at this link

Anonymous Feedback Form

`links.rouxl.es/feedback`

Thanks for coming! 🥳

Please give me feedback on what to improve!