Compsci 101
Pancakes, While loops, Parallel Lists
Live Lecture

Susan Rodger
Nicki Washington
February 25, 2021

Latanya Sweeney

PhD. Computer Science, MIT – first black woman
Over 100 publications, Fellow ACMI

"I am a computer scientist with a long history of weaving technology and policy together to remove stakeholder barriers to technology adoption. My focus is on "computational policy" and I term myself a "computer (cross) policy" scientist. I have enjoyed success at creating technology that weaves with policy to resolve real-world technology-privacy clashes.
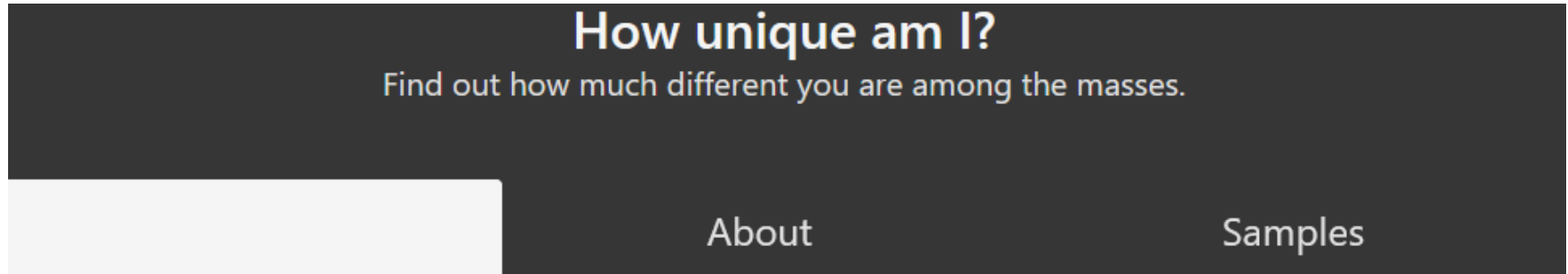
http://latanyasweeney.org/

Identify 87% of US population using (dob,zip,gender). Prof. Government and Technology @ Harvard, instrumental in HIPAA because if *de-identification* work. Former CTO of the Federal Trade Comm.

One of her websites you can try: https://aboutmyinfo.org/identity

How unique am I?
Find out how much different you are among the masses.

About   Samples

Fill out the form below to see how unique you are, and therefore how easy it is to identify you from these values.
Please note that this service is still under development.

Date of Birth — Month ∨ Day ∨ Year ∨

Your Profile

Gender — ⦿ Male ○ Female

ZIP Code — [   ]
ZIP code must be 5 digits long.

Submit →

Results will appear here.

https://aboutmyinfo.org/identity

How unique am I?
Find out how much different you are among the masses.

	About	Samples

Fill out the form below to see how unique you are, and therefore how easy it is to identify you from these values.
Please note that this service is still under development.

Date of Birth [▮] ▼ [▮] ▼ [▮] ▼

Gender ○ Male ◉ Female

ZIP Code [▮]
ZIP code must be 5 digits long.

Submit →

Your Profile

Gender: Female
ZIP Code: [▮] (pop. 46282)

Date of Birth	▮	Easily identifiable by birthdate (about 1).
Birth Year	▮	Lots with your birth year (about 273).
Range	▮ to ▮	Wow! There are lots of people in the same age range as you (about 1365).

Your Profile

Gender: Female

ZIP Code: ████████ (pop. 46282)

Date of Birth	████████	Easily identifiable by birthdate (about 1).
Birth Year	████████	Lots with your birth year (about 273).
Range	████████ to ████████	Wow! There are lots of people in the same age range as you (about 1365).

Five year range

J is for …
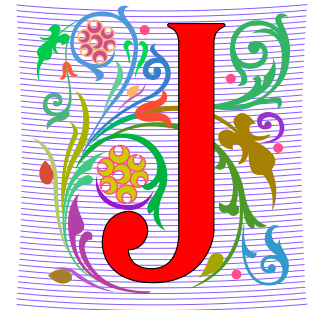
- **JSON**
  - Format for data transmitted across the web
- **JPEG**
  - Image format based on lossy compression
- **Jacquard Loom**
  - 1804 "automated" loom

Announcements

- Nothing due today!
- APT-3 due Tuesday, March 2
- Assignment 2 due Thursday, March 4
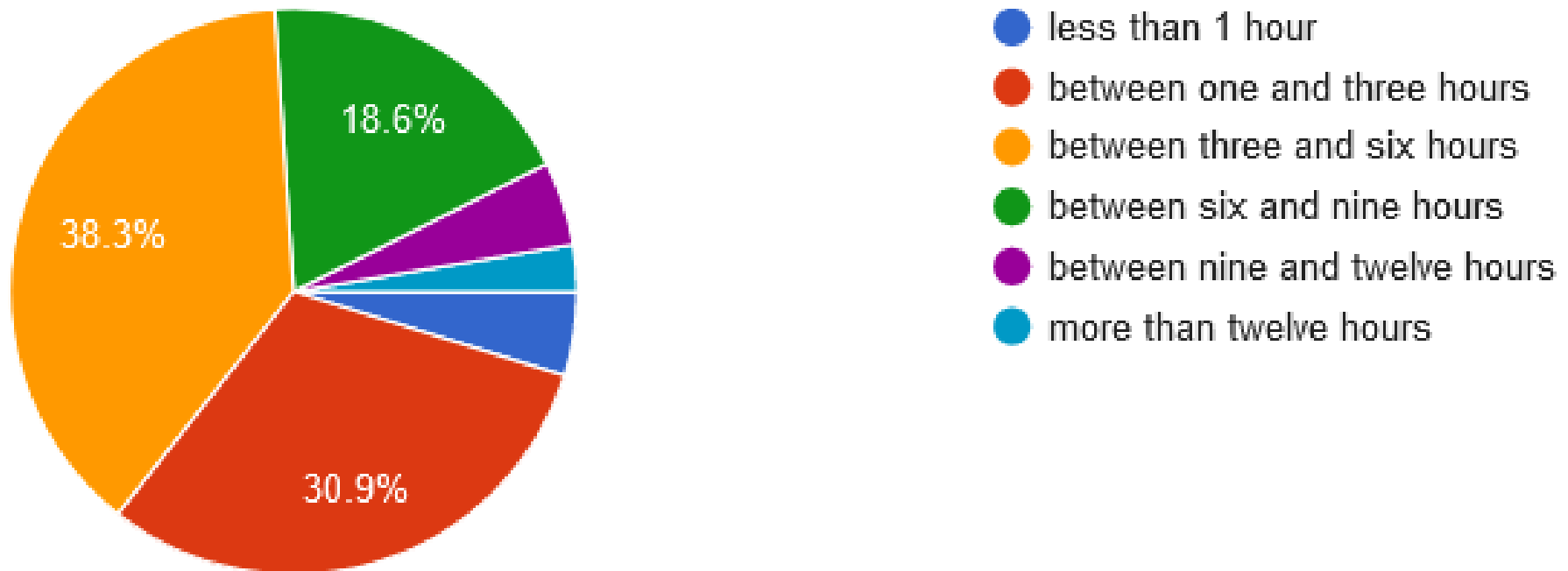- Lab 4 Friday – No Prelab
- APT Quiz 1 3/5-3/8

PFTD

- Files and Data

- Pancake APT

- While loops and Collatz sequence

- Parallel lists

- Exam 1

Exam 1 – Still grading, back soon

- **Once graded, you will get an email from GradeScope**
  - You will be able to see the full exam
  - We will post solutions
    - If you missed something, you should try to figure out what you did, then look at solutions
  - Regrades will be in Gradescope

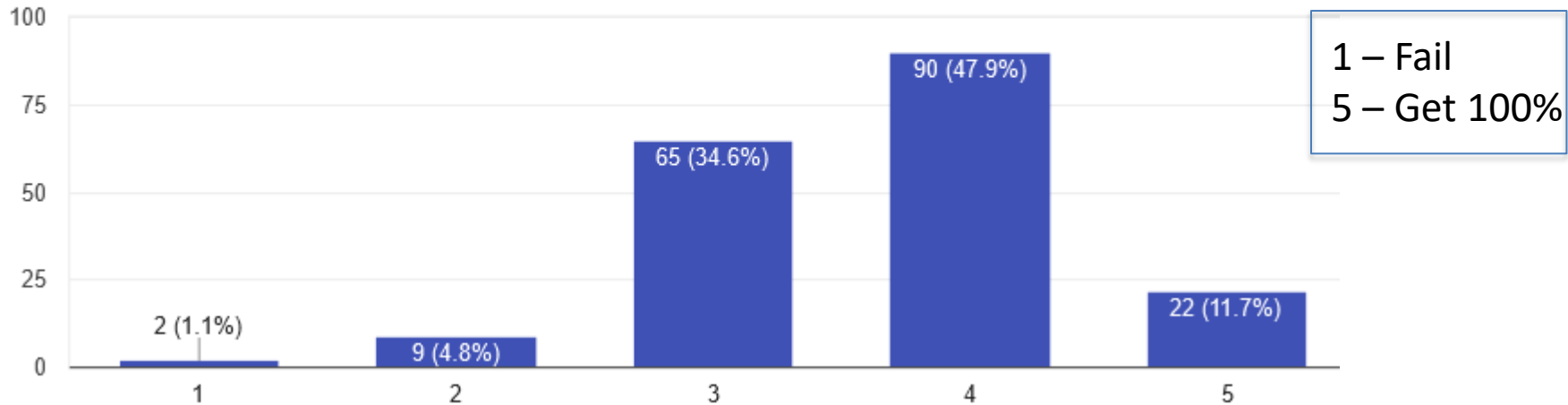 - Here is feedback from the Exam 1 Survey

Exam 1 – How much time preparing?

less than 1 hour
between one and three hours
between three and six hours
between six and nine hours
between nine and twelve hours
more than twelve hours

Thoughts before and after – how you would do/did
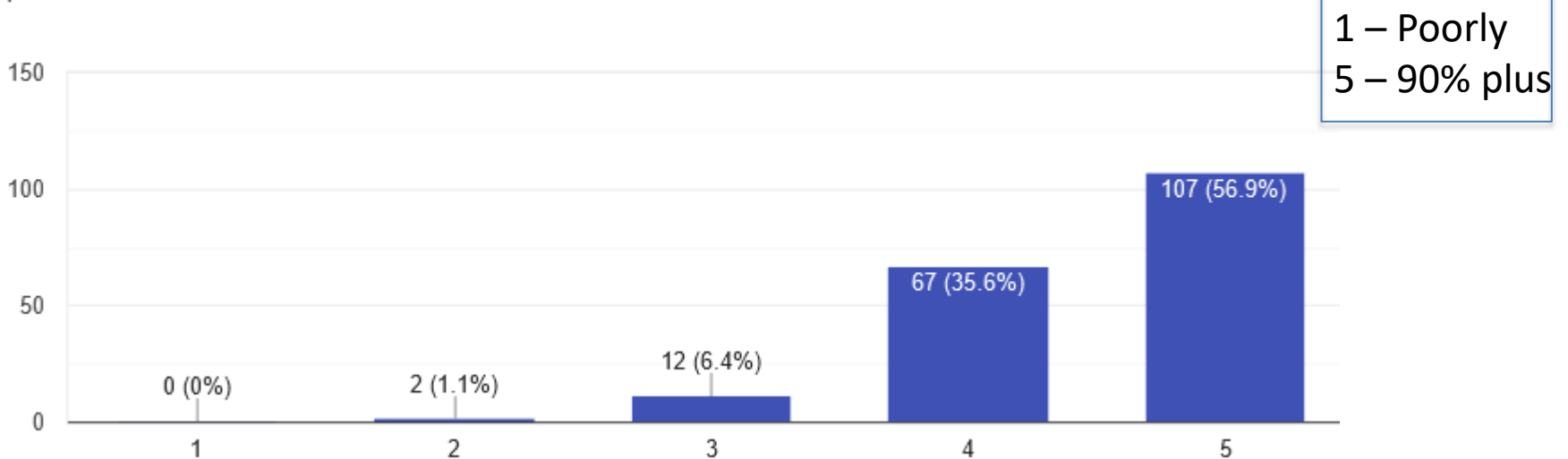
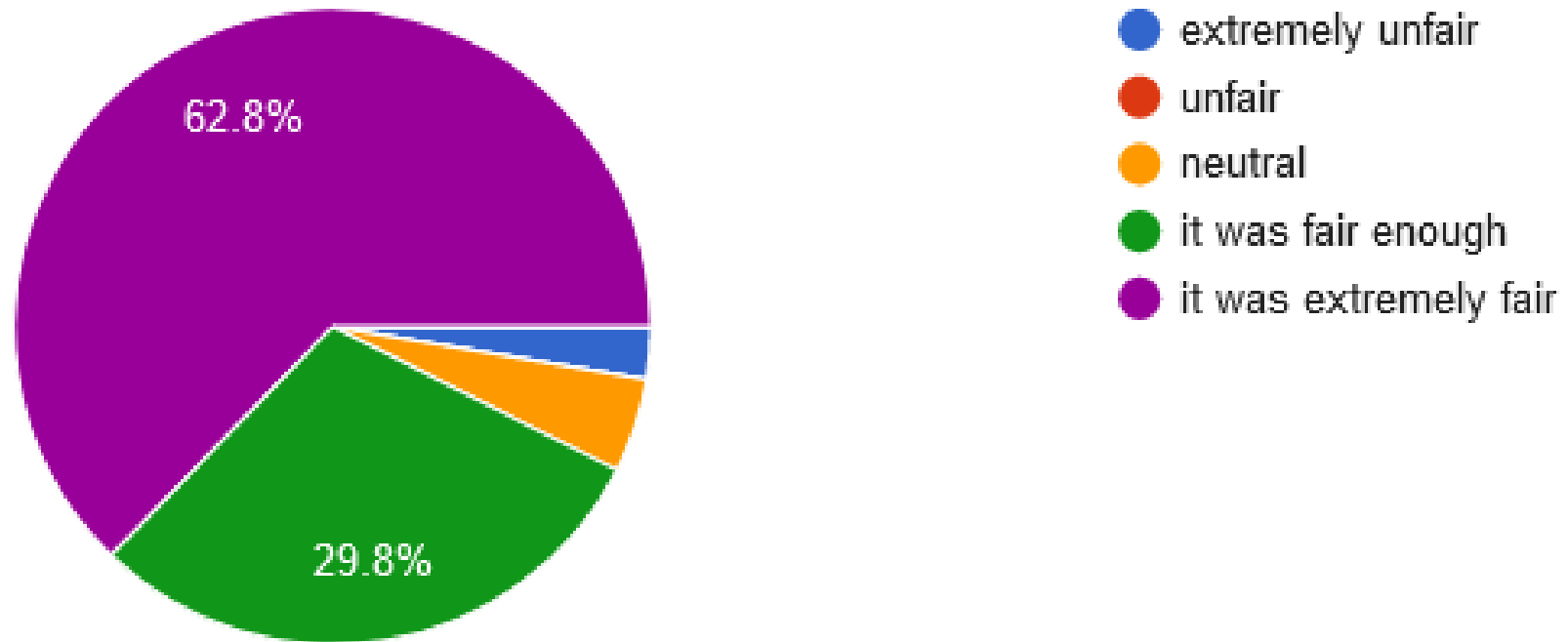As you sat down to start the exam, how did you think you'd do on the exam?

188 responses

1 – Fail
5 – Get 100%

When you completed the exam, how do you think you did?

188 responses

1 – Poorly
5 – 90% plus

Was Exam Fair on questions asked?

62.8%

29.8%

- 🔵 extremely unfair
- 🔴 unfair
- 🟠 neutral
- 🟢 it was fair enough
- 🟣 it was extremely fair
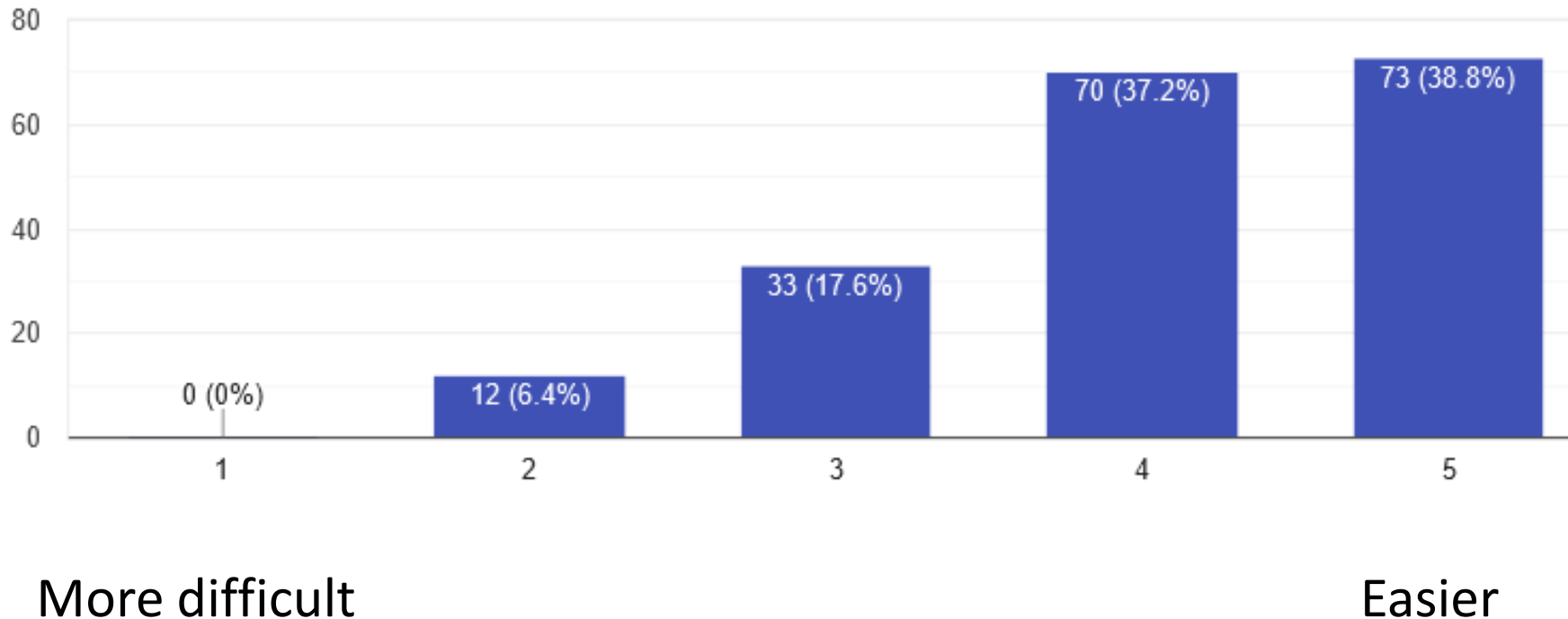
How was taking exam on Gradescope?

How was taking the exam online on gradescope?

188 responses

More difficult   **Easier**
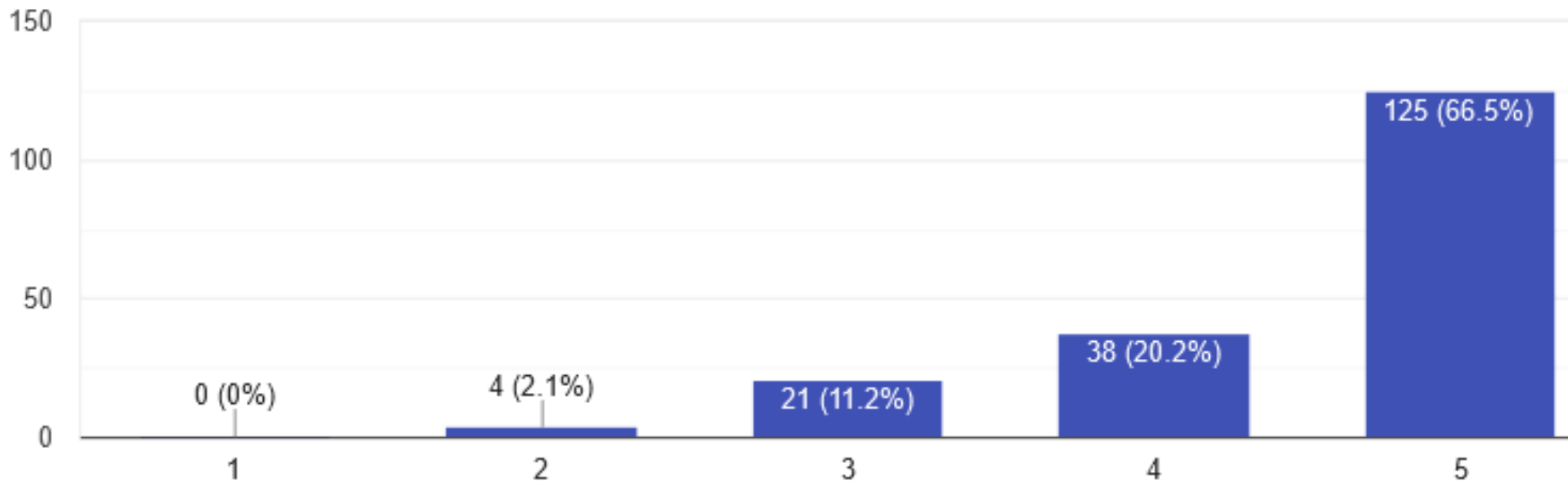
Did you have enough time to take the exam?

Did you have enough time to take the exam?

188 responses

Not Enough time   Plenty of time

APT Quiz 1 coming…

- APT Quiz 1 is 3/5 8AM -3/8 11PM – finish by 11pm
- There are two parts – each part is 1.5 hours
- Pick a start time for each part,
  - Once you start a part, You have 1.5 hours
  - If you get accommodations, you get those
- 4 APTs to solve (2 in each part)
  - Take parts 1 and 2 on same day or different days
- Start APT Quiz on Sakai!
- See old APT Quiz problems so you can practice
  - On APT page – NOT FOR CREDIT

APT Quiz 1

- Is your own work!
  - No collaboration with others!
  - Use your notes, lecture notes, your code, textbook
  - DO NOT search for answers!
  - Do not talk to others about the quiz until grades are posted
- Post private questions on Piazza
  - We are not on between 10pm and 8am EDT!
  - We are not on all the time
  - Will try to answer questions between 8am – 10pm

Lists of Data

- String lists: `["ant","fox","cat","dog"]`
- Lists of int/float numbers: `[5, 3.14159, -15]`
- What about lists of lists? Variable `plist =`
`[["Washington",1789,57],["Clinton",1993,46]]`

- What is `plist[0]`?
- What is `plist[0][2]`?
  - Can always use a variable:

- **First char. of "Washington"?**

Lists of Data

- String lists: `["ant","fox","cat","dog"]`
- Lists of int/float numbers: `[5, 3.14159, -15]`
- What about lists of lists? Variable `plist =`
`[["Washington",1789,57],["Clinton",1993,46]]`

- What is `plist[0]`? ["Washington",1789,57]
- What is `plist[0][2]`? 57
  - Can always use a variable:
  - `val = plist[0]`, then `val[2]`

- **First char. of "Washington"?**
  plist[0][0][0]

WOTO-1 Files
http://bit.ly/101s21-0225-1

Pancakes!

APT Pancake

- **How do you solve this (or any) problem?**
  - 7 Steps!

- **Some APTs are hard problems to solve (step 1-4)**
  - Translating to code easy
- **Some APTs have easy-to-see algorithms (step 5)**
  - Translating to code is hard

APT: Pancakes

Problem Statement

You're a short-order cook in a pancake restaurant, so you need to cook pancakes as fast as possible. You have one pan that can fit `capacity` pancakes at a time. Using this pan you must cook `numCakes` pancakes. Each pancake must be cooked for five minutes on each side, and once a pancake starts cooking on a side it has to cook for five minutes on that side.

Specification

```
filename: Pancakes.py

def minutesNeeded (numCakes, capacity):
    """
    return integer representing time to cook pancakes
    based on integer parameters as described below
    """
```

However, you can take a pancake out of the pan when you're ready to flip it after five minutes and put it back in the pan later to cook it on the other side.

Write the method, `minutesNeeded`, that returns the shortest time needed to cook `numCakes` pancakes in a pan that holds `capacity` pancakes at once. See the examples.

Examples

1.
```
numCakes = 0
capacity = 4

Returns: 0
```

It takes no time to cook 0 pancakes.

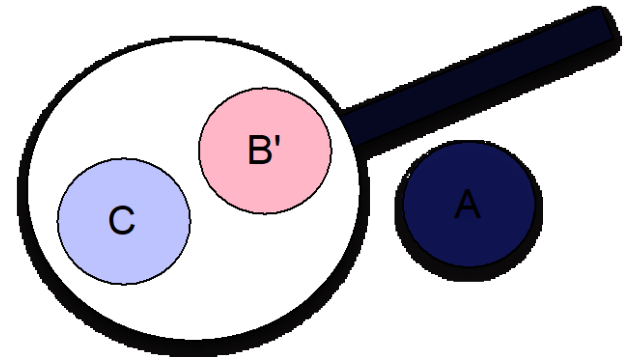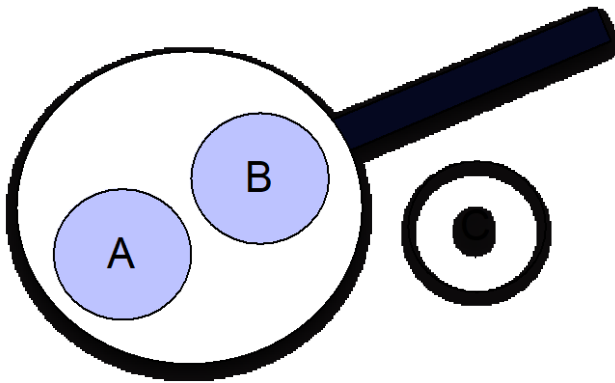2.
```
numCakes = 2
capacity = 2

Returns: 10
```

You cook both pancakes on one side for five minutes, then flip them over and cook each on the other side for another five minutes.

Step 1: Solve an instance
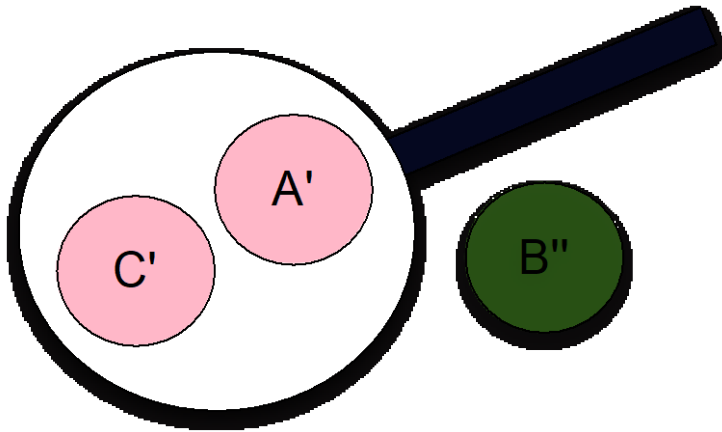## Three pancakes in a two-cake pan

- First 5 minutes
  - 2 half cooking
  - 1 uncooked

- Second 5 minutes
  - 2 half cooking
  - 1 almost cooked

Step 1: Solve an instance
Three pancakes in a two-cake pan

- Third 5 minutes
  - 1 done
  - 2 almost cooked

- How many minutes to cook all three pancakes?

Step 1: Solve an instance

- **What kind of instances? Simple cases that are quickly solved**
  - What are these in Pancake problem?

- **Don't solve for N, solve for 5 (generalize is step 3)**
  - What do when there are two parameters?
    - Fix one, vary the other one
  - Helps identify cases
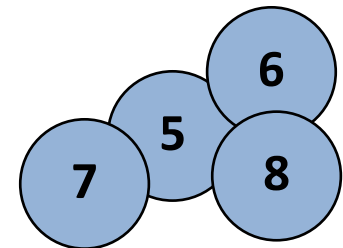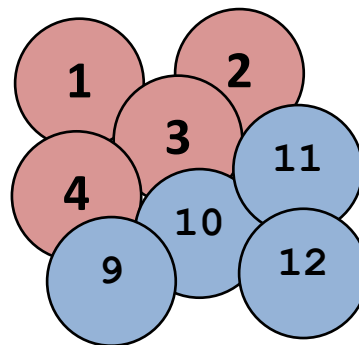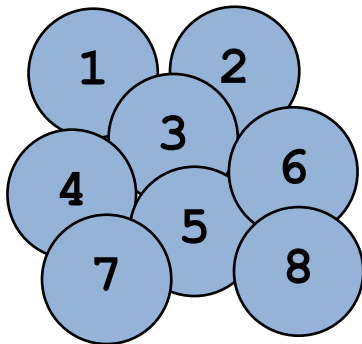
WOTO-2 Pancakes
http://bit.ly/101s21-0225-2

Step 1: Solve an instance

- **Pan has capacity 8, vary # pancakes**
  - Can you cook 12 in 15 minutes? Why?
  - Can you cook 13 in 15 minutes? Why?

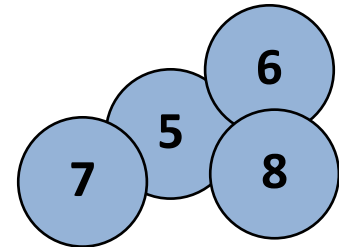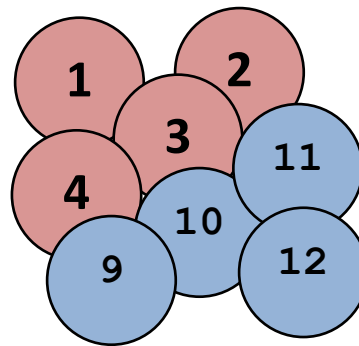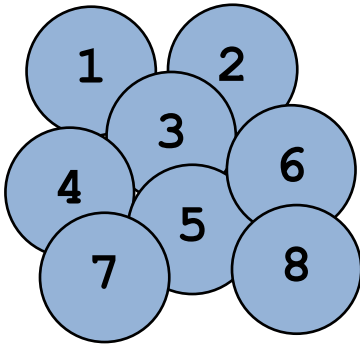| cakes	5	6	7	8	9	10	11	12	13	14	15	16	17	18
time	10	10	10	10	?									

Step 2: What did we just do?

- 13 – 8 = 5
- 8/2 = 4 # Can only take off up to half
- Is 5 <= 4?
  - No, warmer trick won't work
- 10 minutes for 8 pancakes + 10 minutes for 5 more pancakes = 20 minutes

Step 1: Solve an instance

- Pan capacity 8, vary # pancakes, 17 pancakes?

cakes	5	6	7	8	9	10	11	12	13	14	15	16	17	18
time	10	10	10	10	15	15	15	15	20	20	20	20		

# Step 2: What did we just do?

- 17 – 8 = 9, 9 – 8 = 1
- 8/2 = 4
- Is 1 <= 4? # Yes, warmer trick will work!
- Total: 25 minutes
  - 10 minutes for 8 pancakes +
  - 5 minutes for 8 pancakes +
  - Take 1 out, start 17th pancake
  - 5 minutes finish pancakes 8 to 15 +
  - 5 minutes finish pancake 16 and 17

Step 3: Generalize

- Pan has capacity 8, Generalize to algorithm?

cakes	5	6	7	8	9	10	11	12	13	14	15	16	17	18
time	10	10	10	10	15	15	15	15	20	20	20	20	25	25
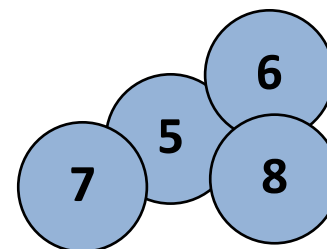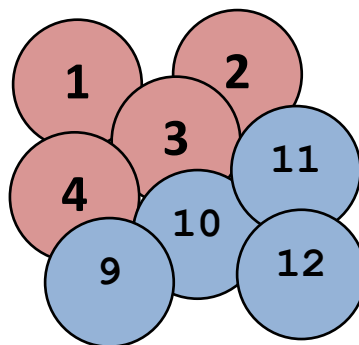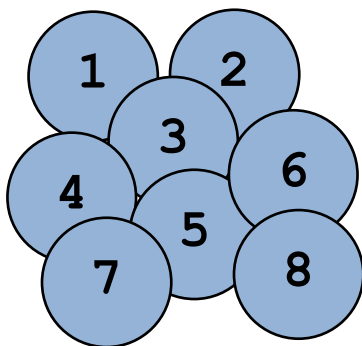
Step 3: Generalize

- 13 – 8 = 5
- 8/2 = 4
- Is 5 <= 4?
  - No, warmer trick won't work
- 10 minutes for 8 pancakes + 10 minutes for 5 more pancakes = 20 minutes

- Remove as many as can with panCapacity
- Will the remainder fit in half the pan?
- Yes, use warmer
  - 5 minutes instead of 10 for last batch
- No, don't use warmer
  - 10 minutes for all sets of panCapacity

Step 4: Test Steps

- Remove as many as can with panCapacity

- Will the remainder fit in half the pan?

- Yes, use warmer

  - 5 minutes instead of 10 for last batch

- No, don't use warmer

  - 10 minutes for all sets of panCapacity

- Case 1:

  - cap 17, cook 34

Step 4: Test Steps

- Remove as many as can with `panCapacity`

- Will the remainder fit in half the pan?

- Yes, use warmer
  - 5 minutes instead of 10 for last batch

- No, don't use warmer
  - 10 minutes for all sets of `panCapacity`

- Case 1:
  - cap 17, cook 34
  - remainder = 0
  - Edge case! No need for warmer
  - Total: 20 minutes

- Case 2:
  - cap 17, cook 42

Step 4: Test Steps

- Remove as many as can with panCapacity
- Will the remainder fit in half the pan?
- Yes, use warmer
  - 5 minutes instead of 10 for last batch
- No, don't use warmer
  - 10 minutes for all sets of panCapacity

- Case 1:
  - cap 17, cook 34
  - remainder = 0
  - Edge case! No need for warmer
  - Total: 20 minutes
- Case 2:
  - cap 17, cook 42
  - remainder = 8
  - Yes, use warmer
  - Total: 25 minutes

Step 5: Code

- Remove as many as can with panCapacity

- Will the remainder fit in half the pan?

- Yes, use warmer
  - 5 minutes instead of 10 for last batch

- No, don't use warmer
  - 10 minutes for all sets of panCapacity

- N pancakes

- How many panCapacity can remove?
  - N // panCapacity

- remainder
  - N % panCapacity

- Half of pan?
  - panCapacity / 2

Let's code it up!

In Summary

- When it comes to planning the algorithm (Steps 1-4) and coding (Step 5), each part can be easy or hard

```python
7   def minutesNeeded(numCakes, capacity):
8       full = numCakes // capacity
9       left = numCakes % capacity
10      minutes = 10 * full
11      if left > capacity/2:
12          minutes += 10
13      elif left > 0:
14          minutes += 5
15      return minutes
```

What are the next steps?

6: Testing!
7: Debugging

Why > and not >=?

How to teach pancake Flipping

- http://www.youtube.com/watch?v=W_gxLKSsSIE
  - For longer, more complex robotic tasks
    - http://www.youtube.com/watch?v=4usoE981e7I

Pancake flipping Video
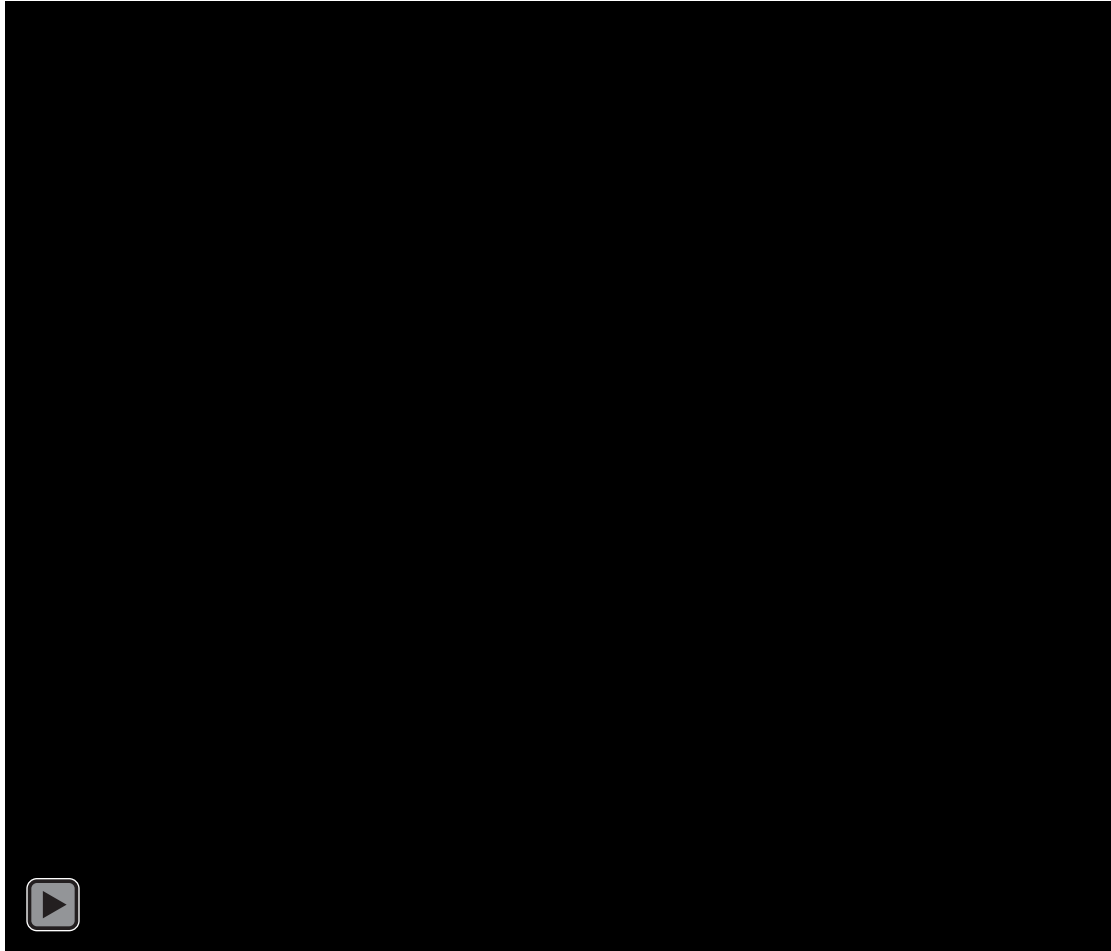
Review: While Loops

- **Don't know:** *how many times* loop executes
  - *a priori* knowledge, we'll know afterword

- **Do know: condition that should be true after loop**
  - Its negation is the expression for `BOOL_CONDITION` (loop guard)

```
while BOOL_CONDITION:
    LOOP_BODY
    # modify variables, affect expression
```

WOTO-3 Collatz and While
http://bit.ly/101s21-0225-3

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named **`words`**
  - Word's count is stored in a list named **`counts`**
  - # occurrences of **`words[k]`** is in **`counts[k]`**

```
["apple", "fox", "vacuum", "lime"]
[5,2,25,15]
```

- What happens when we read a word?

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named `words`
  - Word's count is stored in a list named `counts`
  - # occurrences of `words[k]` is in `counts[k]`

```
["apple", "fox", "vacuum", "lime"]
[5,2,25,15]
```

- What happens when we read a word?

Read word "vacuum"?

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named `words`
  - Word's count is stored in a list named `counts`
  - # occurrences of `words[k]` is in `counts[k]`

```
["apple", "fox", "vacuum", "lime"]
[5,2,26,15]
```

- What happens when we read a word?

Read word "vacuum"?

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named `words`
  - Word's count is stored in a list named `counts`
  - # occurrences of `words[k]` is in `counts[k]`

```
["apple", "fox", "vacuum", "lime"]
[5,2,26,15]
```

- What happens when we read a word?

Read word "cat"?

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named **words**
  - Word's count is stored in a list named **counts**
  - # occurrences of **words[k]** is in **counts[k]**

["apple", "fox", "vacuum", "lime", "cat"]
[5,2,26,15]

Add into words

- What happens when we read a word?

Read word "cat"?

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named `words`
  - Word's count is stored in a list named `counts`
  - # occurrences of `words[k]` is in `counts[k]`

```
["apple", "fox", "vacuum", "lime", "cat"]
[5,2,26,15,0]
```

Expand counts

- What happens when we read a word?

Read word "cat"?

Parallel Lists Review

- We will use parallel lists to track data
  - Each word is stored in a list named **words**
  - Word's count is stored in a list named **counts**
  - # occurrences of **words[k]** is in **counts[k]**

```
["apple", "fox", "vacuum", "lime", "cat"]
[5,2,26,15,1]
```

Add one

- What happens when we read a word?

Read word "cat"?

WOTO-4 File Frequency
http://bit.ly/101s21-0225-4