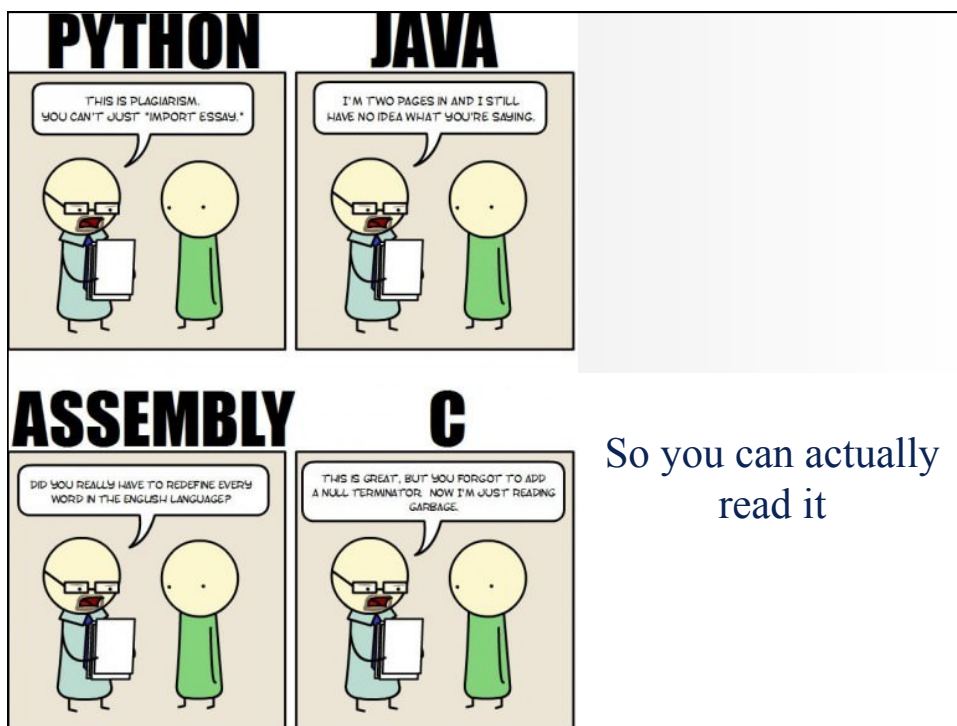




Announcements

- Recitation prep grades
 - Resubmit
- APT set 1 – due tomorrow
- Class code
 - on right-hand side of website
- Lecture videos

APTs

- Make APT java project
 - Add new class for each APT
 - Class / method names EXACTLY match assignment
 - Helper methods can be named anything
 - Test online many times
- Testing/Debugging
 - Output appears in online APT tester
 - Write your own `main`
 - Use the debugger!



3

APT Grading

- Complete required number by due date
- Not keeping up with APTs hurts final APT grade
 - 1/2 of APT grade is completing APTs on time
- “Required” is fair game for exams / recitation
- If you skip an APT you can go back!
 - Keep trying. Hand it in later!

4

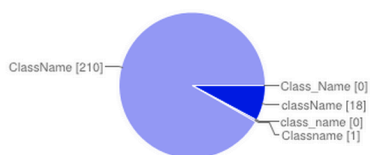
APT Grading

- You can always do more than is required!
 - This will help your grade!
- If you fall behind once - make up for it next time.
 - This won't hurt your grade. (And please don't ask me about it)
- Handing in all APTs at the end of the semester **WILL** hurt your grade!
 - And could hurt your test grades.

5

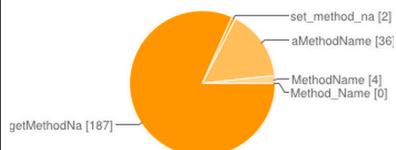
Recitation

Java uses CamelCase for naming variables, methods, and classes. What is a Java appropriate format for naming a class?



className	18	8%
class_name	0	0%
Classname	1	0%
ClassName	210	92%
Class_Name	0	0%

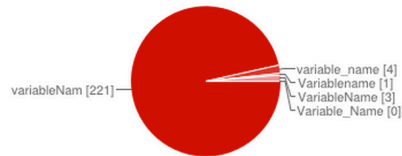
What is a Java appropriate format for naming a method?



getMethodNa	187	82%
set_method_name	2	1%
aMethodName	36	16%
MethodName	4	2%
Method_Name	0	0%

Recitation

What is a Java appropriate format for naming a variable?



variableName	221	97%
variable_name	4	2%
Variablename	1	0%
VariableName	3	1%
Variable_Name	0	0%

Recitation

- Why do we have naming conventions?
 - Reduce effort to read code
 - make graders happy!
 - Enhance code appearance
 - make graders happy!

Recitation

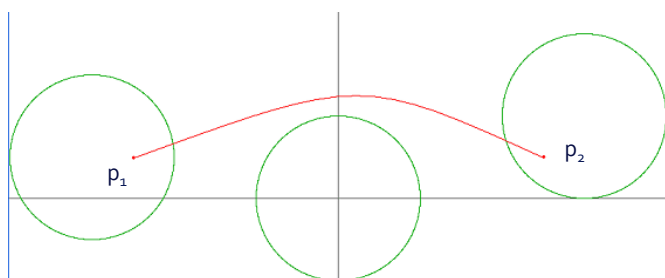
- Write the code to increment through each circle in CirclesCountry

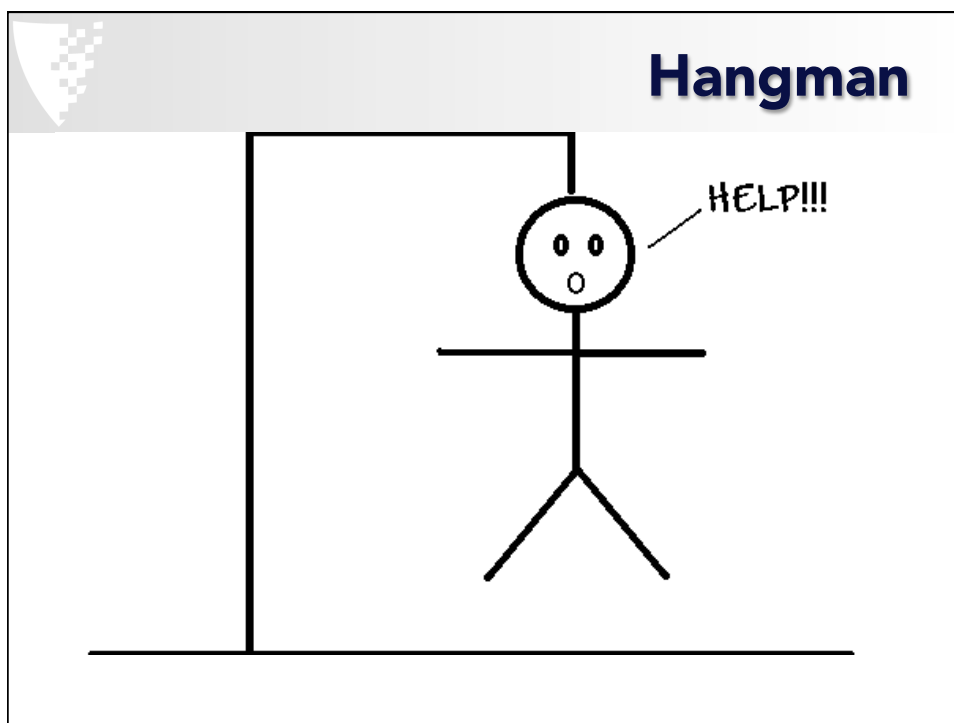
```
public int leastBorders(int[] x, int[] y, int[] r,
                      int x1, int y1, int x2, int y2) {
    // you may have code here

    for(int i = 0; i < x.length; i++){
        //stuff goes in here
    }
}
```

Recitation

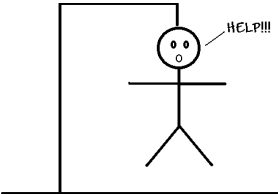
- p_1 and p_2
 - Both in the circle
 - Both out of the circle
 - One in and one out





Hangman

- What to keep track of
 - Secret word
 - Number of misses
 - Letters guessed
- What data structures should you use?



A small version of the Hangman illustration, showing a stick figure hanging from a gallows with a speech bubble saying "HELP!!!".



Arrays

My first data structure
- the organization of data and its
storage allocations in a computer



Some Data Structures

- Array
- List
- Set
- Map

Arrays

- An Array – a collection of items selected by index
 - fixed length
 - single type

```
char[] duke = new char[5];  
duke[0] = 'd';
```



0	1	2	3	4
'd'	'e'	'v'	'i'	'l'

15

Arrays

- How do you create an array of 10 Strings?

```
a. int[] a = new String[10];  
b. String[] b = new String[9];  
c. String[10] c = new String[10];  
d. String[] d = new String[10];
```

16

Arrays

- Fixed length
 - `int[] a = new int[5];`
 - 5 is the length
- type specific
 - `int`, `char`, `String`, `double`, `Dog`, etc.
- Access length of an array
 - `int len = a.length;`

*Note: arrays are Objects - you can make function calls

17

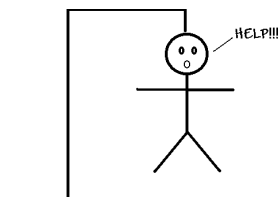
Arrays

- Coding!
 - Create class `ArrayPractive.java`
 - Add method `makeArray` that
 - Creates an array of 50 doubles
 - Puts the number 20.5 into every entry
 - Returns the array

18

Hangman

- What to keep track of
 - Secret word
 - Number of misses
 - Letters guessed

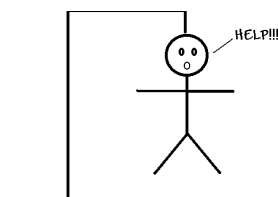


- What data structures should you use?

Hangman

- You have a secret word. How do you save it?

1. `['_', '_', '_', '_'];`
2. `["_", "_", "_", "_"];`
3. `"_ _ _ _";`



- What are the tradeoffs?



Some Data Structures

- Array
- List
- Set
- Map



List

- NOT fixed length

```
ArrayList<String> list = new ArrayList<String>();  
  
// add to ArrayList  
boolean canAdd = list.add("hello");  
  
//check if element is in ArrayList  
boolean inList = list.contains("hello");  
  
//get element at index zero  
String word = list.get(0);
```

List

- No **p**rimitives! Only **O**bjects!

```
ArrayList<int> list = new ArrayList<int>();
```

```
ArrayList<Integer> list = new ArrayList<Integer>();
```

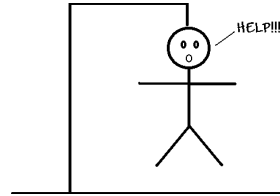


Arrays

- Coding!
 - Modify class `ArrayPractive.java`
 - Add method `makeArray2` that
 - Creates an `ArrayList` of 50 doubles
 - Puts the number 20.5 into every entry
 - Returns the `ArrayList`

Hangman

- What to keep track of
 - Secret word
 - Number of misses
 - Letters guessed



- What data structures should you use?

Some Data Structures

- Array
- List
- Set
- Map