CS106A Winter 2017-18

Final Exam Criteria

**Problem 1: Java of Wakanda (18 points)**

**1a) 4 pts**

-1 minor error

-2 2 minor errors/1 major error

-3 3 minor errors/1 major+minor error

-4 4+ minor errors/2+ major errors

**1b) 5 pts**

What does this program output?

-1 minor error

-2 2 minor errors/1 major error

-3 3+ minor errors/2+ major errors

Give a 1-sentence description of the **isiqingatha** method

-1 minor error

-2 2+ minor errors/1+ major errors

**1c) 4 pts**

-1 minor error

-2 2 minor errors/1 major error

-3 3 minor errors/1 major+minor error

-4 4+ minor errors/2+ major errors

**1d) 5 pts**

Edge cases (a < b, a<= 0, b<= 0)

-1 minor error

-2 2 minor errors/1 major error

-3 3+ minor errors/2+ major errors

Checking divisibility

-1 minor error

-2 2+ minor errors/1+ major error

**Problem 2: Mirror Image (12 points)**

**4 pts** **Loops over pixels**

Outer loop

-1 minor error

-2 2+ minor errors/1+ major error

Inner loop

-1 minor error

-2 2+ minor errors/1+ major error

**4 pts Calculates “mirrored index”**

-1 minor error

-2 2 minor errors/1+major error

-3 3 minor errors/1 major+minor error

-4 4+ minor errors/2+major errors

**2 pts Correctly sets mirrored pixel**

-1 minor error

-2 2+ minor errors/1+ major error

**2 pts** **Returns mirrored image**

-1 minor error

-2 2+ minor errors/1+ major error

**Problem 3: 23 and Character Frequency (18 points)**

**3a) 10 pts**

-1 Fails to create a new **HashMap<Character, Integer>**

-1 Fails to loop over each character in the string

-1 Does not ignore non-alphabetic characters

Does not properly check if the letter has already been seen

-1 minor error

-2 2+ minor errors/1+ major error

Does not add the letter with a count of 1 if not in the HashMap

-1 minor error

-2 2+ minor errors/1+ major error

Does not increment the letter’s count if already in the HashMap

-1 minor error

-2 2+ minor errors/1+ major error

-1 Does not return the HashMap

**3b) 8 pts**

-1 Fails to call **getCharacterCount** for the string to get character frequencies

-1 Fails to loop over each character in the string

-1 Does not ignore non-alphabetic characters

-1 Does not convert the character to uppercase

-1 Does not get the character count from the map

Doesn’t return the character if its frequency is 1

-1 minor error

-2 2+ minor errors/1+ major error

-1 Doesn’t return null if no non-repeating character was found**Problem 4: MPedigree Redux (18 points)**

**2 pts Instance Variables**

Doesn’t create/initialize empty Map/other codes instance variable

-1 minor error

-2 2+ minor errors/1+ major error

**5 pts Reads codes file**

-1 Doesn’t create Scanner

-1 Doesn’t loop over each line in the file

-1 Doesn’t update the codes data structure on each line

-1 Doesn’t close the scanner

-1 Doesn’t print an error message if an error occurs while reading

**8 pts Handles valid “check” requests**

-1 Doesn’t properly detect “check” request

-1 Doesn’t properly get the code parameter from the request

Doesn’t properly check if this code has already been checked

-1 minor error

-2 2+ minor errors/1+ major error

On first time check, fails to update data structure to record check

-1 minor error

-2 2+ minor errors/1+ major error

-1 On first time check, doesn’t return “success”

-1 On subsequent checks, doesn’t return “already checked”

**3 pt Invalid Requests**

-1 Doesn’t return “unknown command” for commands other than “check”

-1 Doesn’t return “missing parameter” if request doesn’t have “code”

-1 Doesn’t return “invalid code” if request’s code is not in codes file

**Problem 5: Infinite Undo (18 points)**

**5 pts Interactors**

-1 Doesn’t create “Undo” button

-1 Doesn’t add “Undo” button

-1 Doesn’t create “Random” button

-1 Doesn’t add “Random” button

-1 Doesn’t add action listeners

**2 pts Instance Variables**

Doesn’t create/initialize empty list/other circles instance variable

-1 minor error

-2 2+ minor errors/1+ major error

**2 pts actionPerformed**

-1 Incorrect method signature

-1 Fails to detect which button is clicked

**5 pts Adding random circles**

-1 Incorrectly sets random coordinates

-1 Incorrectly creates new 50x50 circle

-1 Incorrectly sets random color

-1 Doesn’t add to canvas

-1 Doesn’t record circle in data structure

**4 pts Undo**

-1 Does something/crashes if nothing onscreen

Removes most recent circle from data structure and canvas

-1 minor error

-2 2 minor errors/1 major error

-3 3+ minor errors/2+ major errors

Problem 6: General Card Deck (18 points)

**2 pts Instance Variables**

Doesn’t create/initialize empty list/other cards instance variable

-1 minor error

-2 2+ minor errors/1+ major error

**5 pts Properly implements Constructor**

-1 Doesn’t properly loop over each requested suit

-1 Doesn’t properly loop over each requested value

-1 Doesn’t properly create a new card with the correct suit

-1 Doesn’t properly create a new card with the correct value

-1 Doesn’t properly add the new card to the deck

**3 pts Properly implements getNextCard**

-1 Doesn’t return the empty string if there are no cards left

-1 Doesn’t remove the next card from the deck

-1 Doesn’t return the next card

**2 pts Properly implements getNumRemaining**

-1 minor error

-2 2+ minor errors/1+ major error

**2 pts Properly implements isEmpty**

-1 minor error

-2 2+ minor errors/1+ major error

**4 pts Properly implements shuffle**

-1 doesn’t shuffle for 100 iterations

-1 doesn’t properly generate random indices in the deck

Doesn’t properly swap the cards at the chosen indices

-1 minor error

-2 2 +minor errors/1+ major error**Problem 7: Simple Set (18 points)**

**8 pts Set Validation**

Properly checks if the *suits* make a set (all the same or all different)

-1 minor error

-2 2 minor errors/1 major error

-3 3 minor errors/1 major+minor error

-4 4+ minor errors/2+ major errors

Properly checks if the *values* make a set (all the same or all different)

-1 minor error

-2 2 minor errors/1 major error

-3 3 minor errors/1 major+minor error

-4 4+ minor errors/2+ major errors

**7 pts Gameplay**

-1 Doesn’t loop the program until the deck is empty

Draws and prints the top 3 cards from the deck

-1 minor error

-2 2+ minor errors/1+ major error

-1 Doesn’t properly ask the user whether or not the 3 cards are a set

Validating user answer

-1 minor error

-2 2 minor errors/1 major error

-3 3+ minor errors/2+ major errors

**3 pts Initialization**

Creates a new **GeneralCardDeck**

-1 minor error

-2 2+ minor errors/1+ major errors

-1 Doesn’t shuffle the deck