AP Computer Science
Schedule for Elevens Lab – YOU ARE RESPONSIBLE FOR THESE ASSIGNMENTS IF YOU MISS CLASS TIME

LIVE Tuesday 4/5
Activity One – Exercises 1 and 2

ASYNC Wednesday 4/6
Activity Two – Exercises 1 and 2. Questions 1 – 4.

LIVE Thursday 4/7
Discuss Activities One and Two

ASNYC Friday 4/8
Activity Three – Exercises 1 and 2. Questions 1 – 3
Activity Four – Exercises 1 and 2

LIVE Monday 4/12
Discuss Activities Three and Four

ASYNC Tuesday 4/13
Activity Six – Questions 1 - 3

ASYNC Wednesday 4/14
Activity Seven – Questions 1 – 4

LIVE Thursday 4/15
Discuss activities six and seven

ASYNC Friday 4/16
Activity Eight – Questions 1 – 3

LIVE Monday 4/19
Discuss activity eight

ASYNC Tuesday 4/20
Activity Nine – Exercise 1. Questions 1 - 3

ASYNC Wednesday 4/21
Activity Ten – Exercises 1 – 5

LIVE Thursday 4/22

ASYNC Friday 4/23
Activity Eleven – Exercises 1 – 8. Questions 1 – 3 (not 4!)

Activity One

What should a card class store?

Rank
Suit
Point Value

toString?

NOTES/HINTS:

When testing the equality of two cards (matches), be sure to use .equals and the get methods!

MUST DO

Create Card and CardTester

Activity Two

How would you describe a deck of cards?

Arrays – ranks, suits, point values
ArrayList of cards

NOTES/HINTS:

Remember that these classes are supposed to represent any type of card, not just standard playing cards.

MUST DO

Create Deck and DeckTester
Answer Questions #1-4 on page 6

Activity Three

How would you code the random shuffling of cards?

Arrays – ranks, suits, point values
ArrayList of cards

NOTES/HINTS:

Perfect shuffle pseudocode algorithm on page 8… this is much like which of our sorting algorithms?

Selection shuffle pseudocode algorithm on page 8… this is (obviously) like which of our sorting algorithms?

Why is this second method better, yet inefficient?

MUST DO

Implement perfect shuffle and efficient selection shuffle
Test different values in main
Questions #1-3

Activity Four

Adding the shuffle method to your deck class

NOTES/HINTS:

MUST DO

Exercises 1 and 2 on page 11

(Exercise 2 includes making a standard deck of 52 cards.)

Activity Six (SKIP FIVE!!!)

What is Elevens and how is it played?

NOTES/HINTS:

Just play the game! Easiest assignment ever!

MUST DO

Answer questions #1-3

Activity Seven

How is the ElevensBoard class designed?

NOTES/HINTS:

For question 2, you are only writing the pseudocode!

For question 4c, note the returned array does not need to have the same length as the original!

MUST DO

Questions #1-4 (all parts)

Activity Eight

Using an abstract board class
(Creating other solitaire games like Elevens)

NOTES/HINTS:

"If we use the IS-A test, a ThirteensBoard IS-A ElevensBoard is NOT true. They have a lot in common, but an inheritance relationship between the two does not exist."

"The answer is to use a common superclass!"

MUST DO

Questions #1-3

Activity Nine

Complete the implementation of new ElevenBoards class

NOTES/HINTS:

Be sure your game runs correctly!

The cards directory must be in the same directory

MUST DO

Exercises #1(a-d)

Questions #1-3

Activity Ten

Create ThirteensBoard using what you have learned

NOTES/HINTS:

Do not reinvent the wheel!

MUST DO

Exercises #1-5

Activity Eleven (#irony)

Creating a simulation of our games rather than playing them ourselves.

NOTES/HINTS:

MUST DO

Exercises #1-8
Questions #1-3 (skip #4)