Logic-1 Quiz
http://codingbat.com/home/srp4379@lausd.net/quiz3

1. gotAdvisory, Hint: 3 squirrelPlay

2. diceDoublesOrLucky7, Hint: *24 withoutDoubles*, Any of the problems from *09 specialEleven* thru *13 nearTen*

3. digitsEqualOrTwice, Hint: 16 teaParty, 22 lastDigit, 29 shareDigit

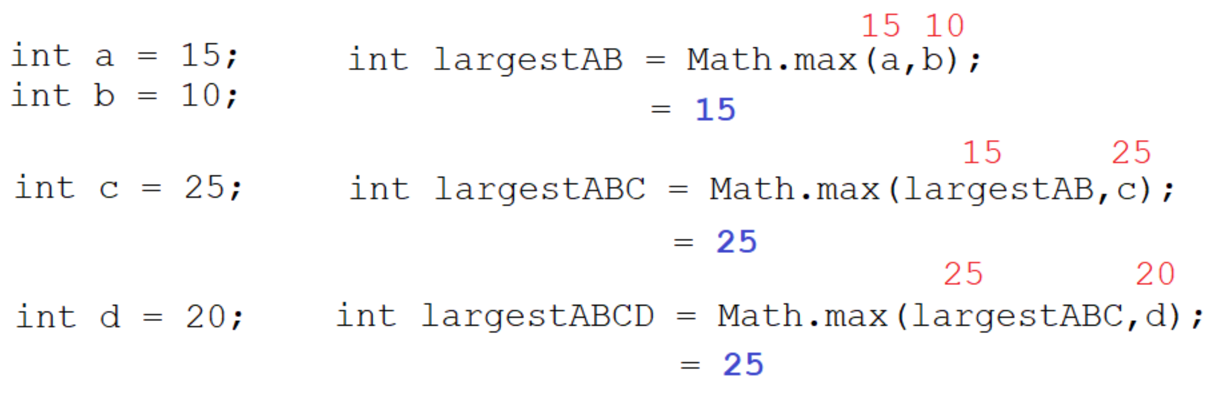
4. consecutiveOrder, Hint: 20 inOrder, 21 inOrderEqual,
HINT: Note that the numbers must not only be IN ORDER,
but each number must be exactly 1 more than the previous number.
For an inorder sequence like 4 8 11, 4 < 8 && 8 < 11.
This is also true for a consecutive sequence like 3 4 5: 3 < 4 && 4 < 5.
HOWEVER, for the consecutive sequence this ALSO must be true:
3 + 1 == 4 && 4 + 1 == 5

5. **oddSum**, Hint: 5 sortaSum, Logic-1 Basics boolean **isEvenNumber**

6. **bothEvenOrBothOdd**, Hint: Logic-1 Basics boolean **isEvenNumber, isDivisibleBy3and5 !!**

7. divisibleBy5, Hint: Logic-1 Basics **isDivisibleBy3**

8. largestOf3



```
                                        15  10
int a = 15;     int largestAB = Math.max(a,b);
int b = 10;                   = 15
                                            15       25
int c = 25;     int largestABC = Math.max(largestAB,c);
                               = 25
                                           25        20
int d = 20;     int largestABCD = Math.max(largestABC,d);
                                = 25
```

9. subjectNow

10. middleOf3
HINT: There are two possible solutions:
#1:
If you have three numbers: **n1 n2** and **n3**,
if **n1** is the middle number, then it can be in the middle in 2 ways:
n2 <= **n1** && **n1** <= **n3** (**n2** is low and **n3** is high)
OR **n3** <= **n1** && **n1** <= **n2** (**n3** is low and **n2** is high)
#2:
You already know how to find the **largest** of 3 numbers using **Math.max**().
You can find the **smallest** of 3 numbers using **Math.min**().
Consider that the **total** of the 3 numbers is the sum: **n1 + n2 + n3**
Note that the **largest + smallest + middle** must ALSO total to that same sum**.**
Therefore the **middle value** must be equal to the **total - (largest + smallest)!**