Quiz 1:

#1. What is the approximate depth of a Decision Tree trained (without restrictions) on a training set with 1 million instances? (Assume that the tree is well-balanced and makes only binary decisions) [3min]

A. 10
B. 16
C. 20
D. 1 million
Correction: 20

#2. Is a node's Gini impurity: [2min]
a. generally lower
b. always lower
c. generally greater
b. always greater
Correction: a. generally lower

#3. Consider a node containing 4 instances of class A and 1 of class B. What is the Gini impurity of this node? [4min]
A. 0.32
B. 0
C. 1.6
D. 0.23
Correction: A

#4. Consider a node containing 4 instances of class A and 1 of class B. Then, suppose the dataset is one-dimensional and the instances are lined up in the following order: A, B, A, A, A. Compute the Gini index child values if you split this node: [8 min]
a. After the first A (i.e.: A, SPLIT, B, A, A, A)
b. After B (i.e.: A, B, SPLIT, A, A, A)
c. After the second A (i.e.: A, B, A, SPLIT, A, A)
d. After the third A (i.e.: A, B, A, A, SPLIT, A)
Give these four couple values in the following order: (Gini index value of the left child, Gini index value of the right child) and precise for each couple the associated split (a. b. c. or d.).
(For example (this example is a fake): split a. (0 , 0.1); split b. (0.2,0.3); split c. (0.4,0.5); split d. (0.6, 0.7))
Correction: split a. (0, 0.375); split b. (0.5, 0); split c. (0.44, 0); split d. (0.375, 0)

#5. Based on your previous computing, what is the best split? (Help: for each split, compute the weighted Gini impurity named « CART cost function for classification » in course 1.). [4 min]
A. The split a.
B. The split b.
C. The split c.
D. The split d.
Correction: B. (Because the weighted Gini impurity values are 0.3 for the split a., 0.2 for the split b., 0.26 for the split c., 0.3 for the split d.).

#6. Thanks to these previous computing, do you think you rightly answer to the question 2 (reminder: the question 2 was « A node's Gini impurity is « generally lower » or « generally greater » or « always lower » or « always greater » than its parent's?) [1 min]
[No correction : funny question]

#7. If a Decision Tree is overfitting the training set, is it a good idea to try decreasing the value of the max_depth hyperparameter (or stopping criterion)? [2 min]
A. TRUE
B. FALSE
Correction: True

8. When a Decision Tree is overfitting, why it's a good idea to decrease the value of max_depth? [2 min]
Correction: If a Decision Tree is overfitting the training set, it may be a good idea to decrease max_depth, since this will constrain the model, regularizing it.

#9. Can you give another hyper parameter that could help to avoid overfitting by tuning its value? [2 min]
Correction: maximum number of sample per leaf, or, maximum number of leaves

#10. If a Decision Tree is underfitting the training set, is it a good idea to try scaling the input features? [2 min]
A. TRUE
B. FALSE
Correction: B. False. Decision Trees don't care whether or not the training data is scaled or centered; that's one of the nice things about them. So, if a Decision Tree underfits the training set, scaling the input features will just be a waste of time.