CIS 472/572 Homework #1 (Written)

Mamtaj Akter

TOTAL POINTS

25.5 / 30

QUESTION 1

1 Question 1 8.5 / 10

+ **2.5 pts** 1.a Correct

✓ **+ 2.5 pts** **1.b Correct**

✓ **+ 2.5 pts** **1.c Correct**

✓ **+ 2.5 pts** **1.d Correct**

✓ **+ 1 pts** **1.a The answers of B node are swapped**

+ **2 pts** 1.a There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1.5 pts** 1.a At A=0, the B expansion is unnecessary.

+ **1 pts** 1.a Structure is correct. All answers are wrong.

+ **1.5 pts** 1.b Answer for A=False and B=False missing

+ **2 pts** 1.b There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1.5 pts** 1.b A=T should go to "Yes"

+ **1.5 pts** 1.b At A=False,B=False Answer should be "No".

+ **1.5 pts** 1.b Both branches of C are labeled as 1.

+ **1.5 pts** 1.b At A=True, B=True, C=False. The label is missing.

+ **1.5 pts** 1.b One branch label of A is missing.

+ **1 pts** 1.b Multiple redundancies.

+ **0.5 pts** 1.c Structure is correct all answers are wrong.

+ **2 pts** 1.c There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1 pts** 1.c The sub tree at A=False is wrong.

+ **1 pts** 1.c At A=T, the expansion on B is missing.

+ **2 pts** 1.d There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1.5 pts** 1.d At A=No, C=Yes both of your D branches are labeled "No".

+ **1.5 pts** 1d At A=F and C=T branch, D has been left unexpanded.

+ **1.5 pts** 1.d One branch label of D is missing at A=0, C=1.

+ **1.5 pts** 1.d At A=T,B=F the C=T branch label is missing.

+ **1.5 pts** 1.d On the "A=true? no" branch, expanding on B is irrelevant.

+ **1 pts** 1.d Multiple redundancies.

+ **1 pts** 1.d At A=True,B=False,C=False the expansion of D is unnecessary.
At A=False, C=False the expansion of D is unnecessary.

QUESTION 2

Question 2 10 pts

2.1 Question 2a 2 / 2

✓ **+ 2 pts** **Correct**

+ **0 pts** The labels are overcast, rain, and sunny.

+ **1 pts** Overcast is an edge label not a variable.

2.2 Question 2b 2 / 2

✓ **+ 2 pts** **Correct**

+ **0 pts** Not answered.

+ **1 pts** Data points have not been assigned to the overcast branch.

+ **0 pts** Data points have not been assigned to the root's branch labels.

+ **1 pts** Datapoint D14 has not been assigned

+ **1 pts** Datapoint D4 has not been assigned

+ **1.5 pts** D1 assigned to "rain" in-place of D4

D4 assigned to "sunny" in-place of D1

+ **0.5 pts** Only the assignment Sunny is correct.

+ **1.5 pts** "Sunny" has not been labeled

+ **1.5 pts** D11 not assigned.

2.3 Question 2c **6 / 6**

✓ **+ 2 pts** **Sunny branch is correct**

✓ **+ 2 pts** **Overcast branch is correct**

✓ **+ 2 pts** **Rain branch is correct**

+ **0 pts** Overcast branch is missing

+ **1 pts** The leaf value of the overcast branch is not mentioned.

+ **1.5 pts** The leaf value of the overcast branch should be "yes".

+ **1.5 pts** Sunny branch is correct. However, calculations are not shown adequately.

+ **1.5 pts** Rain branch is correct. However, calculations are not shown adequately.

+ **1 pts** Rain branch is correct. However, calculations are done for Gain(Rain,Humidity)

+ **0 pts** Sunny branch is missing

+ **0 pts** Rain branch is missing

+ **0 pts** Sunny branch calculations are wrong.

+ **0 pts** Rain branch calculations are wrong.

+ **1.5 pts** Sunny branch calculations are correct but the selected node is not mentioned.

+ **1.5 pts** Overcast branch calculations are correct but the selected leaf node value is not mentioned.

+ **1.5 pts** Rain branch calculations are correct but the selected node is not mentioned.

+ **0.5 pts** Sunny branch calculations are wrong. However the selected node is correct.

+ **0.5 pts** Rain branch calculations are wrong. However the selected node is correct.

QUESTION 3

Question 3 10 pts

3.1 Question 3a **3 / 6**

+ **6 pts** Correct

+ **0 pts** Your answer is what the bank is doing now. You were asked to combine the trees.

✓ **+ 3 pts** **What happens when the tress have more than 2 levels? (i.e: multiple "low risk" leaves and multiple "high risk" leaves)**

+ **2 pts** The algorithm is to simply replace the leaves of one decision tree with a low-risk label by the root of the other tree.

+ **2 pts** Append where?

3.2 Question 3b **4 / 4**

✓ **+ 4 pts** **Correct**

+ **2 pts** Very good that you have noticed the n1-1 condition. However the correct answer is simple n1*n2. This is because when the O notation (upper bound is used) the lower polynomials and constants vanish.

+ **1 pts** You are assuming that exactly half of the leaves of each tree will be low risk and the other half be high risk. This is not the correct worst case scenario.

+ **1 pts** For the upper bound, you can suppose that all leaves of the first tree is low-risk. (Because we are taking O notation) Then the upper bound is n1*n2.

+ **1 pts** For the upper bound, you can suppose that all leaves of the first tree is low-risk. Then the upper bound is n1*n2. (Adding extra variables is not correct. The answer should be based on the given parameters which are n1 and n2.)

+ **1 pts** if n1^k are the leaves that get replaced, then the leaf nodes on those branches will be n1^k*n2
Also, since n1^r=(n1 -n1^k) using the n1^r notation is redundant.

Now, the nodes are given by n_1^k*n2+(n1-n1^k)
Solve above
n1^k (n2-1)+n1
Since we are using the big O notation
it becomes n1^k *n2
(the lower polynomials and constants vanish.)
What is the upperbound for n1^k now? it is n1.

Thus the final upper-bound is the simple: n1*n2

+ **1.5 pts** Very good that you have noticed the n1-1

condition.
There is no reason to take n2-1 because all of its leaves will stay as leaves.
Further, the correct answer is simple n1*n2. This is because when the O notation (upper bound is used) the lower polynomials and constants vanish.

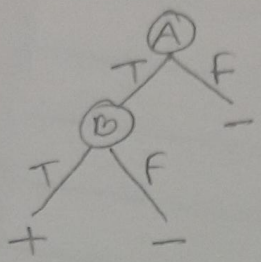
+ **1 pts** You are correct to think that it will be the multiplication n1*n2
BUT you cannot go ahead and simplify it to n^2 because there is no mention that n1=n2. So you should have stopped at n1*n2.

+ **1.5 pts** Good that you have noticed the n1-T1 condition.
Since we are using upper bound, T1=1.
Then, the correct answer is simple n1*n2. This is because when the O notation (upper bound is used) the lower polynomials and constants vanish.

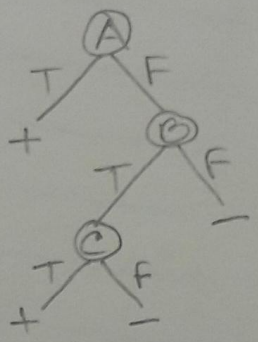
CIS 572, Winter 2018
Homework 1 (Written): Decision Trees

UO ID: 951641885
Name: Mamtaj Akter

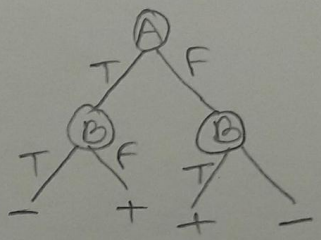
1. a) $A \wedge \neg B$



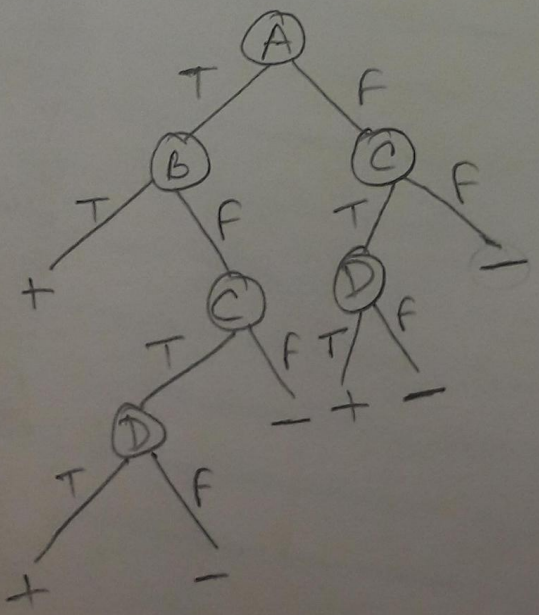
b) $A \vee [B \wedge C]$



c) A XOR B



d) $[A \wedge B] \vee [C \wedge D]$



1 Question 1 8.5 / 10

+ **2.5 pts** 1.a Correct

✓ **+ 2.5 pts** **1.b Correct**

✓ **+ 2.5 pts** **1.c Correct**

✓ **+ 2.5 pts** **1.d Correct**

✓ **+ 1 pts** **1.a The answers of B node are swapped**

+ **2 pts** 1.a There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1.5 pts** 1.a At A=0, the B expansion is unnecessary.

+ **1 pts** 1.a Structure is correct. All answers are wrong.

+ **1.5 pts** 1.b Answer for A=False and B=False missing

+ **2 pts** 1.b There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1.5 pts** 1.b A=T should go to "Yes"

+ **1.5 pts** 1.b At A=False,B=False Answer should be "No".

+ **1.5 pts** 1.b Both branches of C are labeled as 1.

+ **1.5 pts** 1.b At A=True, B=True, C=False. The label is missing.

+ **1.5 pts** 1.b One branch label of A is missing.

+ **1 pts** 1.b Multiple redundancies.

+ **0.5 pts** 1.c Structure is correct all answers are wrong.

+ **2 pts** 1.c There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1 pts** 1.c The sub tree at A=False is wrong.

+ **1 pts** 1.c At A=T, the expansion on B is missing.

+ **2 pts** 1.d There is no indication to decide whether to take left or right branch when a node is true or false.

+ **1.5 pts** 1.d At A=No, C=Yes both of your D branches are labeled "No".

+ **1.5 pts** 1d At A=F and C=T branch, D has been left unexpanded.

+ **1.5 pts** 1.d One branch label of D is missing at A=0, C=1.

+ **1.5 pts** 1.d At A=T,B=F the C=T branch label is missing.

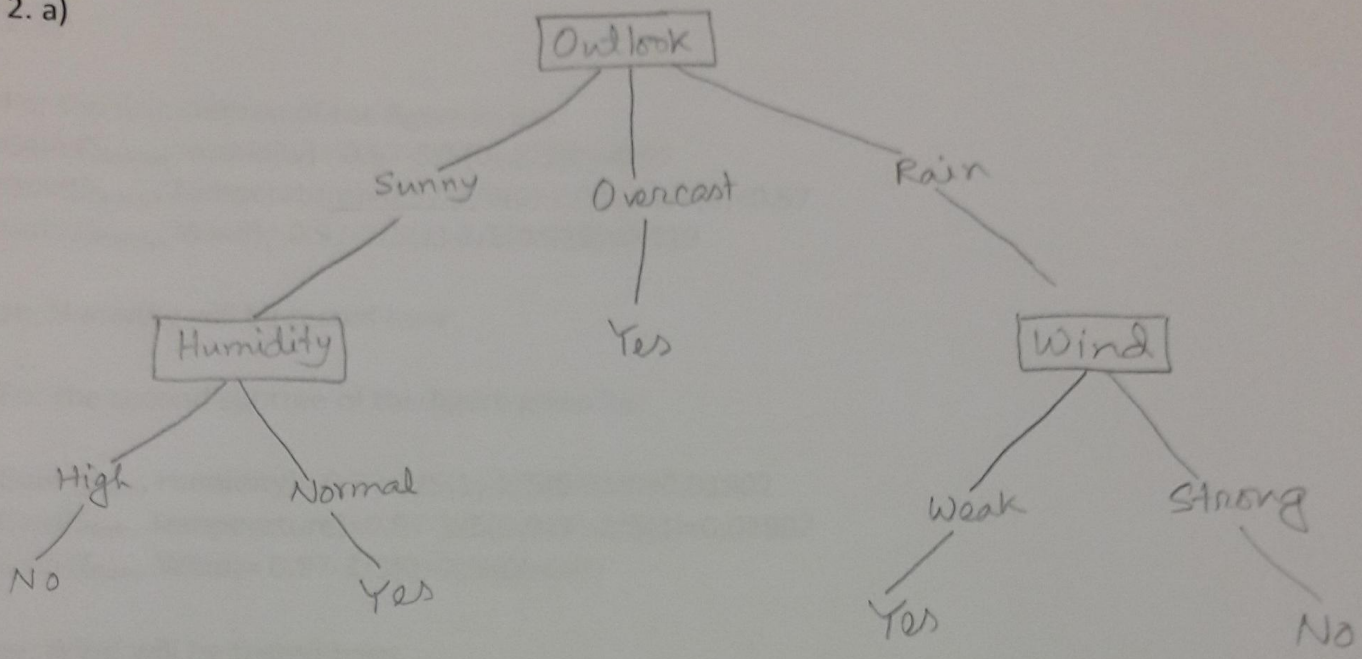
+ **1.5 pts** 1.d On the "A=true? no" branch, expanding on B is irrelevant.

+ **1 pts** 1.d Multiple redundancies.

+ **1 pts** 1.d At A=True,B=False,C=False the expansion of D is unnecessary.

At A=False, C=False the expansion of D is unnecessary.

2. a)



b)

OUTLOOK: SUNNY	D1,D2,D8,D9,D11
OUTLOOK: OVERCAST	D3,D7,D12,D13
OUTLOOK: RAIN	D4,D5,D6,D10,D14

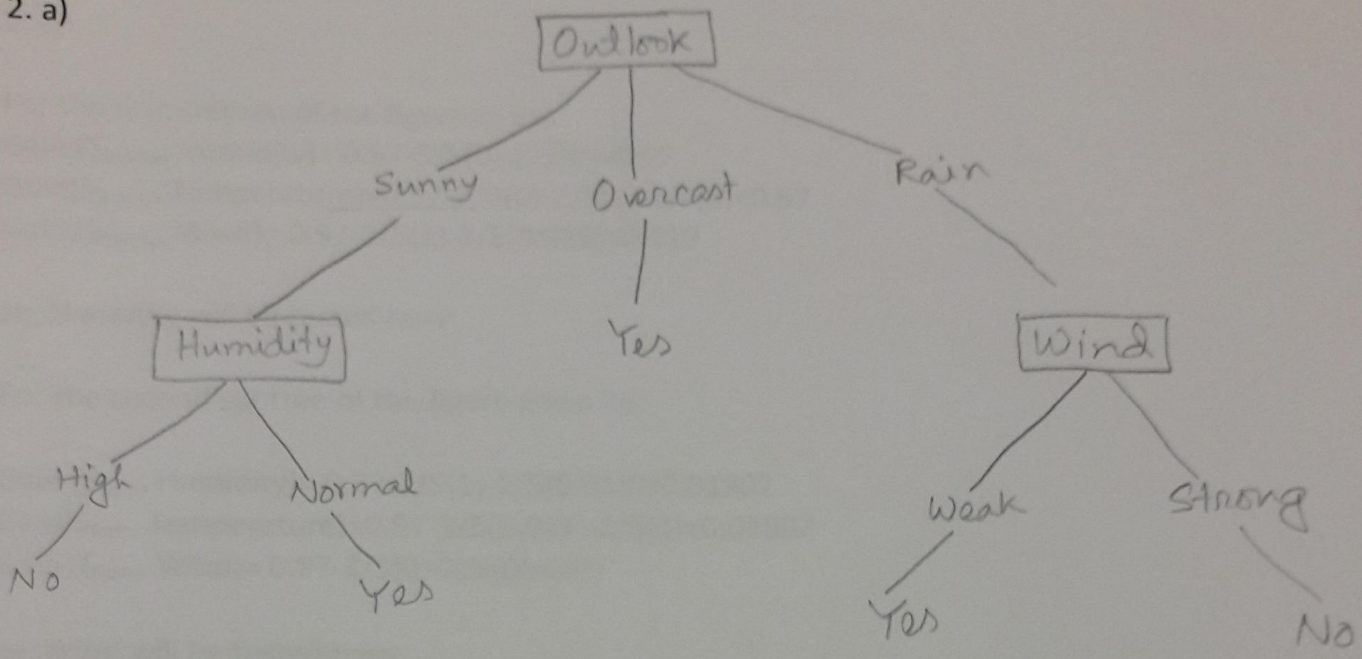
2.1 Question 2a 2 / 2

✓ **+ 2 pts Correct**

+ 0 pts The labels are overcast, rain, and sunny.

+ 1 pts Overcast is an edge label not a variable.

2. a)



b)

OUTLOOK: SUNNY	D1,D2,D8,D9,D11
OUTLOOK: OVERCAST	D3,D7,D12,D13
OUTLOOK: RAIN	D4,D5,D6,D10,D14

2.2 Question 2b 2 / 2

✓ **+ 2 pts Correct**

+ 0 pts Not answered.

+ 1 pts Data points have not been assigned to the overcast branch.

+ 0 pts Data points have not been assigned to the root's branch labels.

+ 1 pts Datapoint D14 has not been assigned

+ 1 pts Datapoint D4 has not been assigned

+ 1.5 pts D1 assigned to "rain" in-place of D4

D4 assigned to "sunny" in-place of D1

+ 0.5 pts Only the assignment Sunny is correct.

+ 1.5 pts "Sunny" has not been labeled

+ 1.5 pts D11 not assigned.

2. c)

For the first subtree of the figure in 2a:
Gain (S_{Sunny}, Humidity)= 0.97-3/5(0)-2/5(0)=0.97
Gain(S_{Sunny}, Temperature)=0.97-2/5(0)-2/5(1)-1/5(0)=0.57
Gain (S_{Sunny}, Wind)= 0.97-2/5(1)-3/5(0.918)=0.019

so, Humidity will be tested here.

For the second subtree of the figure given 2a:

Gain (S_{Rain}, Humidity)= 0.97-2/5(1)-2/5(0.918)=0.01902
Gain(S_{Rain}, Temperature)=0.97-3/5(0.918)-2/5(1)=0.01902
Gain (S_{Rain}, Wind)= 0.97-3/5(0)-2/5(0)=0.97

so, Wind will be tested here.

For Outlook is Overcast, there will be no branch taken as the decision is "Yes" always.

2.3 Question 2c 6 / 6

✓ **+ 2 pts** **Sunny branch is correct**

✓ **+ 2 pts** **Overcast branch is correct**

✓ **+ 2 pts** **Rain branch is correct**

+ 0 pts Overcast branch is missing

+ 1 pts The leaf value of the overcast branch is not mentioned.

+ 1.5 pts The leaf value of the overcast branch should be "yes".

+ 1.5 pts Sunny branch is correct. However, calculations are not shown adequately.

+ 1.5 pts Rain branch is correct. However, calculations are not shown adequately.

+ 1 pts Rain branch is correct. However, calculations are done for Gain(Rain,Humidity)

+ 0 pts Sunny branch is missing

+ 0 pts Rain branch is missing

+ 0 pts Sunny branch calculations are wrong.

+ 0 pts Rain branch calculations are wrong.

+ 1.5 pts Sunny branch calculations are correct but the selected node is not mentioned.

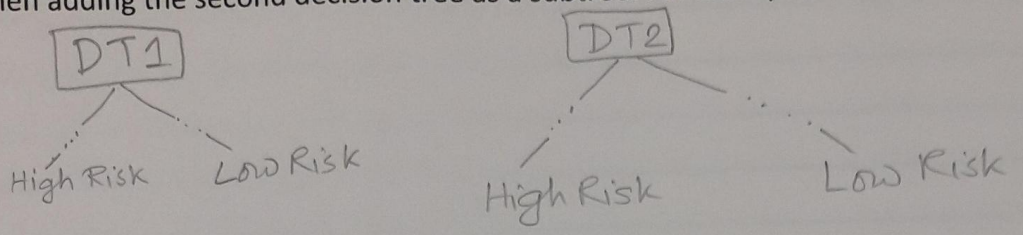
+ 1.5 pts Overcast branch calculations are correct but the selected leaf node value is not mentioned.

+ 1.5 pts Rain branch calculations are correct but the selected node is not mentioned.

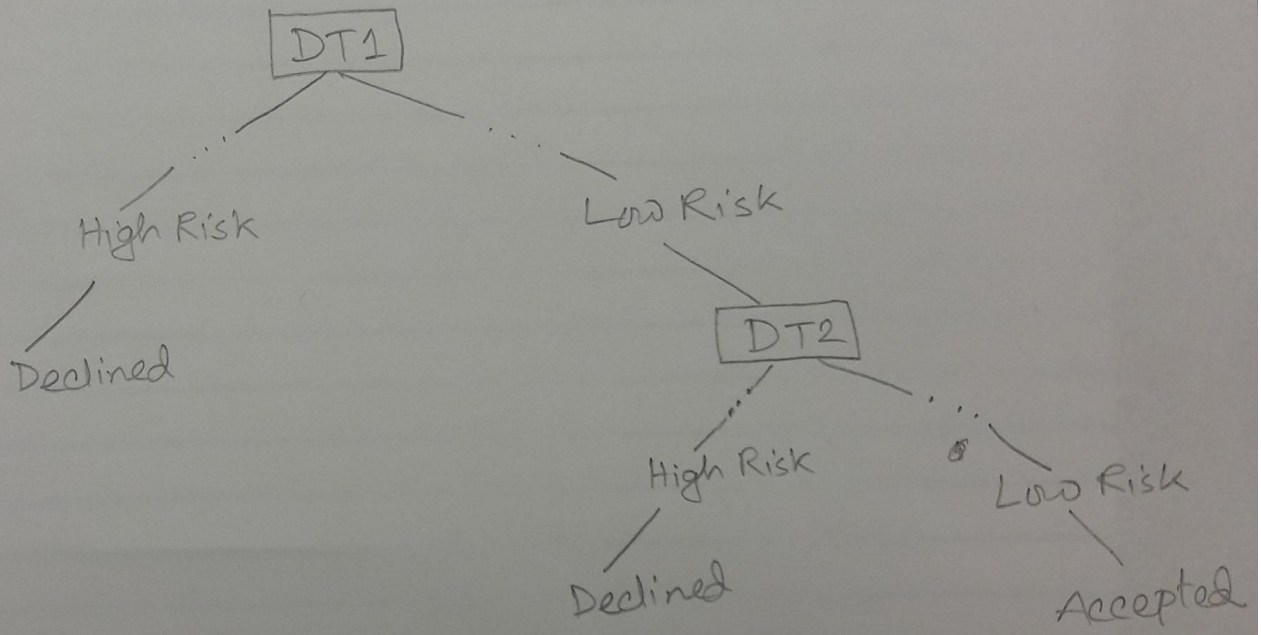
+ 0.5 pts Sunny branch calculations are wrong. However the selected node is correct.

+ 0.5 pts Rain branch calculations are wrong. However the selected node is correct.

3. a) To merge these two decision trees, we can pick any one tree first and start parsing and then adding the second decision tree as a subtree. For example:



To convert them into one single decision tree, lets pick DT1 first.



3. b) the upper bound of n of the the merged tree can be n1Xn2. So, it is $O(n_1 n_2)$.

3.1 Question 3a **3 / 6**

+ **6 pts** Correct

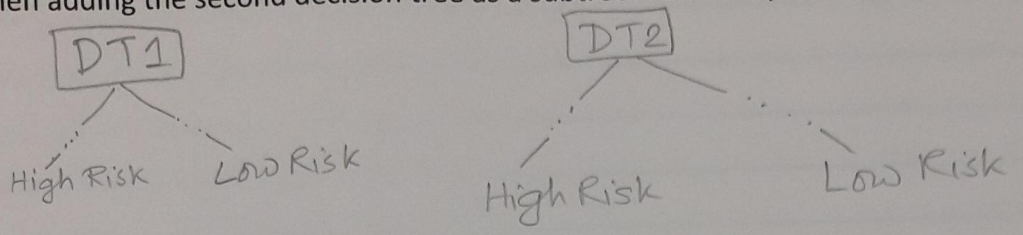
+ **0 pts** Your answer is what the bank is doing now. You were asked to combine the trees.

✓ **+ 3 pts** **What happens when the tress have more than 2 levels? (i.e: multiple "low risk" leaves and multiple "high risk" leaves)**

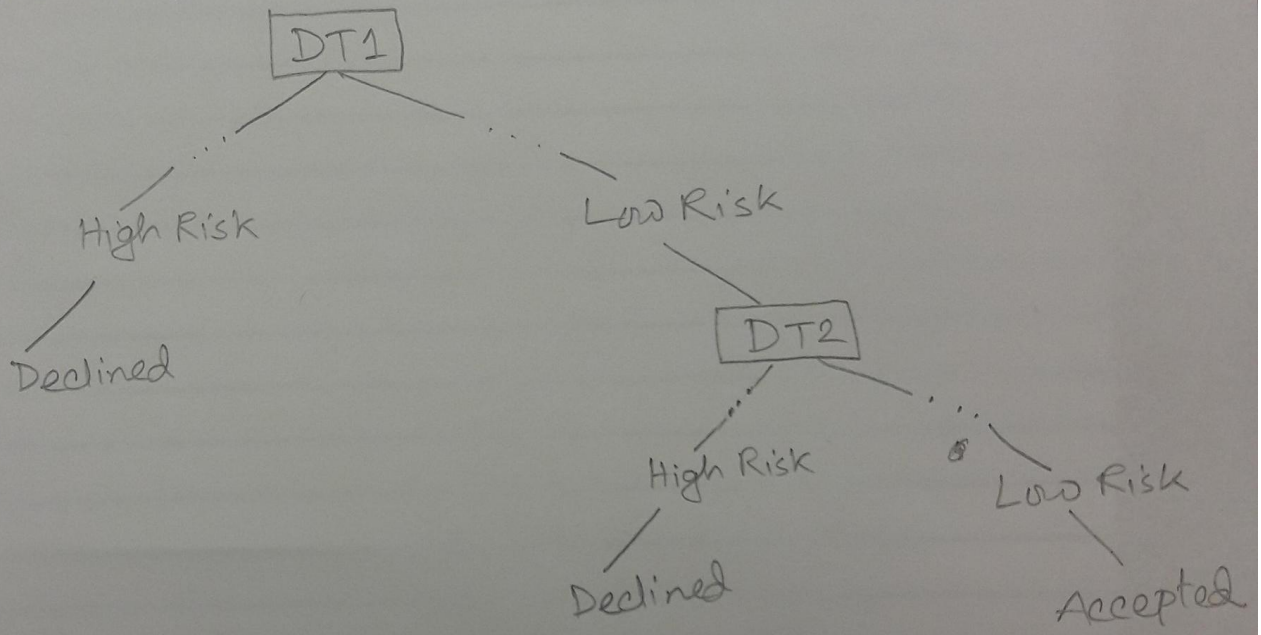
+ **2 pts** The algorithm is to simply replace the leaves of one decision tree with a low-risk label by the root of the other tree.

+ **2 pts** Append where?

3. a) To merge these two decision trees, we can pick any one tree first and start parsing and then adding the second decision tree as a subtree. For example:



To convert them into one single decision tree, lets pick DT1 first.



3. b) the upper bound of n of the the merged tree can be n1Xn2. So, it is $O(n_1 n_2)$.

3.2 Question 3b 4 / 4

✓ **+ 4 pts Correct**

+ 2 pts Very good that you have noticed the n1-1 condition. However the correct answer is simple n1*n2. This is because when the O notation (upper bound is used) the lower polynomials and constants vanish.

+ 1 pts You are assuming that exactly half of the leaves of each tree will be low risk and the other half be high risk. This is not the correct worst case scenario.

+ 1 pts For the upper bound, you can suppose that all leaves of the first tree is low-risk. (Because we are taking O notation) Then the upper bound is n1*n2.

+ 1 pts For the upper bound, you can suppose that all leaves of the first tree is low-risk. Then the upper bound is n1*n2. (Adding extra variables is not correct. The answer should be based on the given parameters which are n1 and n2.)

+ 1 pts if n1^k are the leaves that get replaced, then the leaf nodes on those branches will be n1^k*n2
Also, since n1^r=(n1 -n1^k) using the n1^r notation is redundant.

Now, the nodes are given by n_1^k*n2+(n1-n1^k)
Solve above
n1^k (n2-1)+n1
Since we are using the big O notation
it becomes n1^k *n2
(the lower polynomials and constants vanish.)
What is the upperbound for n1^k now? it is n1.

Thus the final upper-bound is the simple: n1*n2

+ 1.5 pts Very good that you have noticed the n1-1 condition.
There is no reason to take n2-1 because all of its leaves will stay as leaves.
Further, the correct answer is simple n1*n2. This is because when the O notation (upper bound is used) the lower polynomials and constants vanish.

+ 1 pts You are correct to think that it will be the multiplication n1*n2
BUT you cannot go ahead and simplify it to n^2 because there is no mention that n1=n2. So you should have stopped at n1*n2.

+ 1.5 pts Good that you have noticed the n1-T1 condition.
Since we are using upper bound, T1=1.
Then, the correct answer is simple n1*n2. This is because when the O notation (upper bound is used) the lower polynomials and constants vanish.