Name	
email	

6.034 Final Examination
December 15, 2010

Circle your TA and principle recitation instructor so that we can more easily identify with whom you have studied:

Martin Couturier	Kenny Donahue	Gleb Kuznetsov
Kendra Pugh	Mark Seifter	Yuan Shen

Robert Berwick	Randall Davis	Lisa Fisher

Indicate the approximate percent of the lectures, mega recitations, recitations, and tutorials you have attended so that we can better gauge their correlation with quiz and final performance and with attendance after OCW video goes on line. Your answers have no effect on your grade.

	Lectures	Recitations	Megas	Tutorials
Percent attended				

Quiz	**Score**	**Grader**
Q1		
Q2		
Q3		
Q4		
Q5		

There are 48 pages in this final examination, including this one. In addition, tear-off sheets are provided at the end with duplicate drawings and data. As always, open book, open notes, open just about everything.

Quiz 1, Problem 1, Rule Systems (50 points)

Kenny has designed two suits for the Soldier Design Competition, and he and Martin like to grapple in the prototypes on Kresge Oval.

- Kendra insists the suits qualify as **“deadly weapons”** and Kenny should give them to her for safekeeping.
- Kenny and Martin insist that they are examples of an **“enhanced prosthesis”** and that they should be able to keep them

The TAs decide to use Rule-Based Systems to resolve their dispute.

Rules:

P0	IF (AND ('(?x) is a Crazy Physicist', '(?x) is an Engineer') THEN ('(?x) builds a Weaponized Suit'))
P1	IF ('(?y)'s (?x) is Cardinal Red' THEN ('(?y)'s (?x) is not US Govt. Property'))
P2	IF (OR (AND ('(?y) is an Engineer', '(?y)'s (?x) is Really Heavy'), '(?y)'s (?x) is stolen by the Air Force') THEN ('(?y)'s (?x) is a Deadly Weapon'))
P3	IF (OR ('(?y) is not evil', '(?y) is a Roboticist') THEN ('(?y)'s suit research is not Evil'))
P4	IF (AND ('(?y)'s (?x) research is not Evil', '(?y)'s (?x) is not US Govt. Property') THEN ('(?y)'s (?x) is an Enhanced Prosthesis'))

Assertions:

A0: (Kenny is a Roboticist)

A1: (Martin is an Engineer)

A2: (Kenny's suit is Cardinal Red)

A3: (Martin's suit is Really Heavy)

Part A: Backward Chaining (30 points)

Make the following assumptions about backward chaining:

- The backward chainer tries to find a matching assertion in the list of assertions. If no matching assertion is found, the backward chainer tries to find a rule with a matching consequent. In case none are found, then the backward chainer assumes the hypothesis is false.
- The backward chainer never alters the list of assertions; it never derives the same result twice.
- Rules are tried in the order they appear.
- Antecedents are tried in the order they appear.

Simulate backward chaining with the hypothesis

Kenny's suit is an enhanced prosthesis

Write all the hypotheses the backward chainer looks for in the database in the order that the hypotheses are looked for. The table has more lines than you need. We recommend that you use the space provided on the next page to draw the goal tree that would be created by backward chaining from this hypothesis. **The goal tree will help us to assign partial credit** in the event you have mistakes on the list.

1	Kenny's suit is an enhanced prosthesis
2	
3	
4	
5	
6	
7	
8	
9	
10	

Draw Goal Tree Here for Partial Credit

Part B: Forward Chaining (20 points)

Let's say, instead, our assertions list looked like this:

A0: Gleb is an Engineer
A1: Gleb's laptop is Really Heavy
A2: Gleb's suit is Really Heavy

B1 (4 points)

CIRCLE any and all rules that match in the first iteration of forward chaining

P0	P1	P2	P3	P4

B2 (4 points)

What assertion(s) are added or deleted from the database, as a consequence of this iteration?

B3 (4 points)

CIRCLE any and all rules that match in the second iteration of forward chaining

P0	P1	P2	P3	P4

B4 (4 points)

What assertion(s) are added or deleted from the database, as a consequence of this iteration?

B5 (4 points)

You take the same assertions as at the beginning of problem B, above, and re-order them:

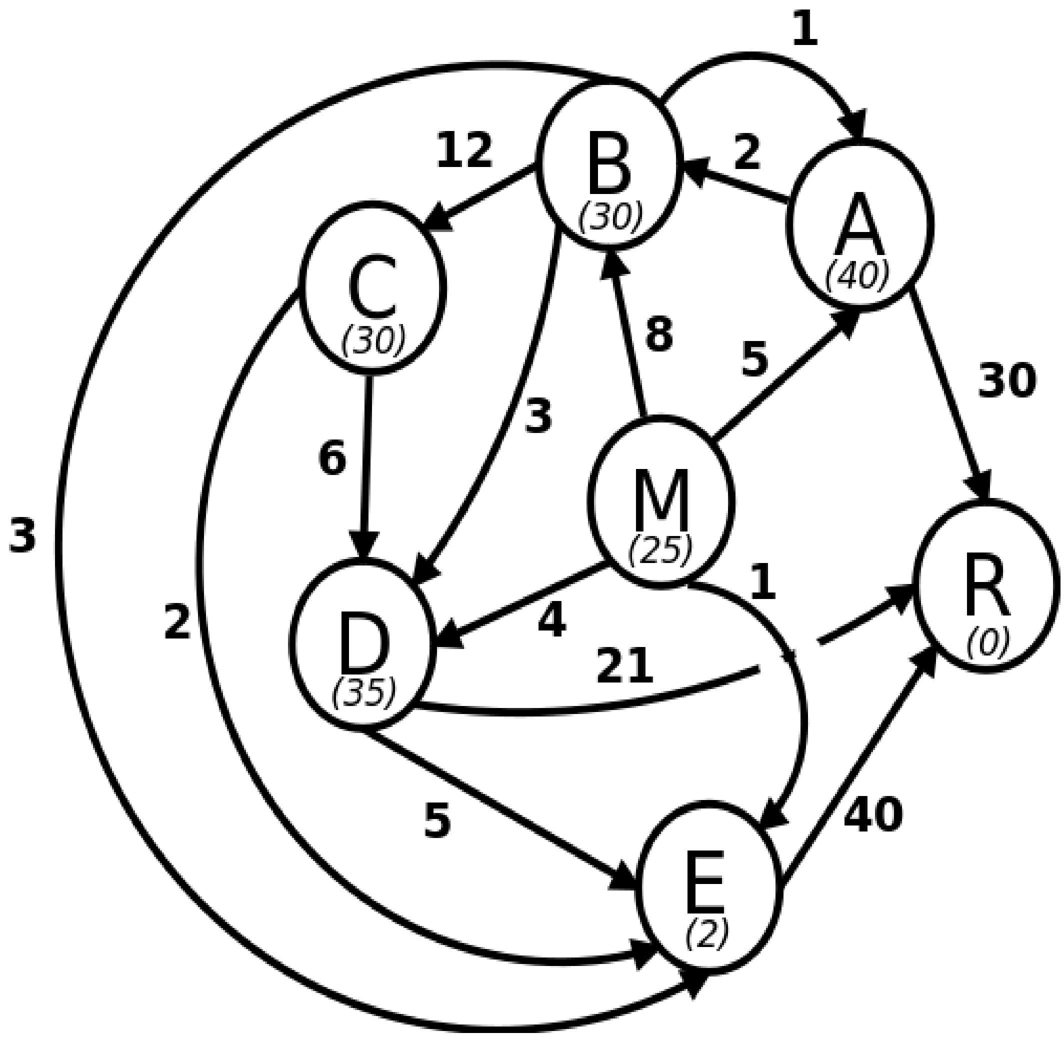
A0: Gleb is an Engineer
A1: Gleb's suit is Really Heavy
A2: Gleb's laptop is Really Heavy

Now, you start over, and run forward chaining from the beginning, until no new assertions are added to or deleted from the database. Is Gleb's laptop a Deadly Weapon?

Quiz 1, Problem 2, Search (50 points)

As you get close to graduating MIT, you decide to do some career planning. You create a graph of your options where the start node is **M = MIT** and your goal node is **R = Retire**, with a bunch of options in between. Your graph includes edge distances that represent, roughly, the "cost of transition" between these careers (don't think too hard about what this means). You also have heuristic node-to-goal distances which represent your preconceptions about how many more years you have to work until you retire. For example, you think it will take 25 years to go from MIT (M) to retirement (R), 30 years from Grad School (B), but only 2 years from Entrepreneur (E).

A = Wall Street | B = Grad School | C = Professor | D = Government | E = Entrepreneur



Part A: Basic search (25 points)

In all search problems, use alphabetical order to break ties when deciding the priority to use for extending nodes.

A1 (3 points)

Assume you want to retire after doing the least number of different jobs. Of all the basic search algorithms you learned about(that is, excluding branch and bound and A*) which one should you apply to the graph in order to find a path, **with the least search effort**, that has the minimum number of nodes from M to R?

A2 Basic Search Chosen Above (7 points)

Perform the search you wrote down in A1 (with an Extended List). Draw the search tree and give the final path.

Tree:

Path:

A3 Beam Search with w=2 (15 points)

Now you are interested in finding a path and the associated distance. Try a **Beam Search** with a width **w=2**, ***with*** an extended list. As before, you are looking for a path from **M** to **R**. Use the "preconceptions" heuristic distances indicated in parentheses at each node.

Tree:

Path, if any:

Extended nodes in order extended:

Part B: Advanced Search (25 points)

B1 Branch and Bound with Extended List (15 points)

Use Branch and Bound search with an Extended List to find a path from **M** to **R,** as well as the extended node list. Use this space to draw the corresponding tree and show your work.

Tree:

Path:

Extended nodes in order extended:

B2 Thinking about Search (9 points)

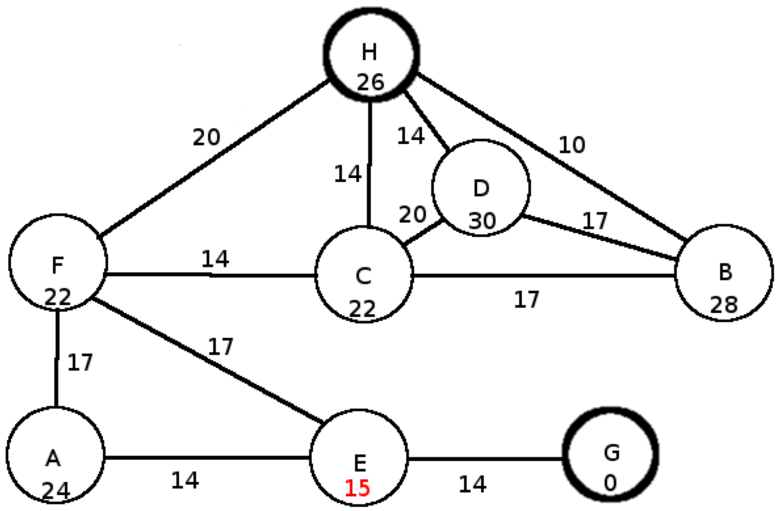
Concisely explain why Branch and Bound with Extended List yields a different result than Beam Search in this problem.

What can we say about the path found by Branch and Bound with Extended List? (We're looking for a fairly strong statement here.)

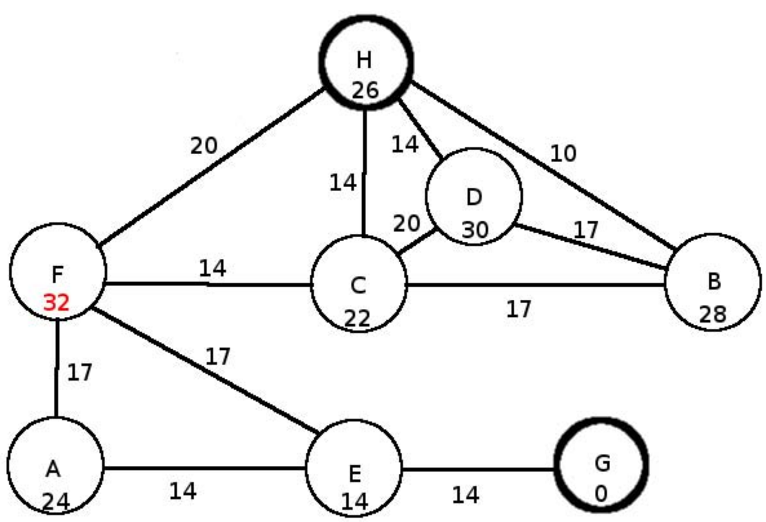
Is there an algorithm that guarantees the same answer as Branch and Bound ***for the graph in this problem***, but can find the answer with fewer extended paths. If Yes, what is that algorithm? If No, explain why not.

B3 Permissible Heuristics (6 points)

Suppose you are asked to find the shortest path from H to G in the graphs below. For both of the graphs explain why the heuristic values shown are not valid for A*. Note the differences in the graphs at nodes F and E.



Reason(s):



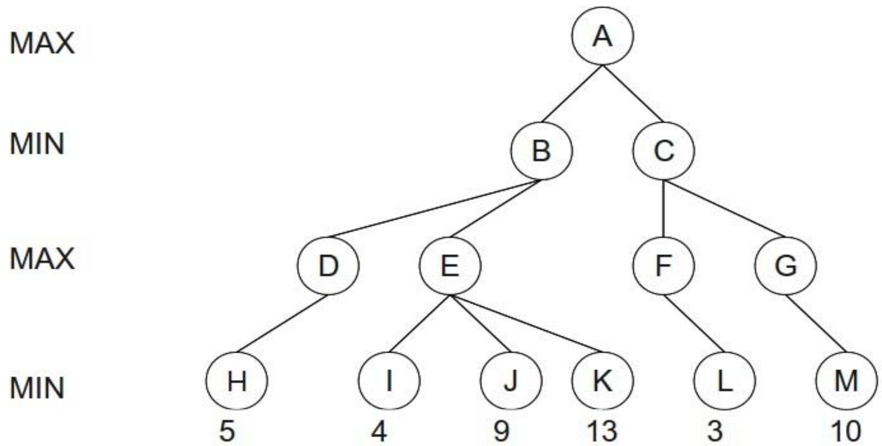
Reason(s):

Quiz 2, Problem 1, Games (50 points)

Part A: Basics(15 points)

A1 Plain Old Minimax(7 points)

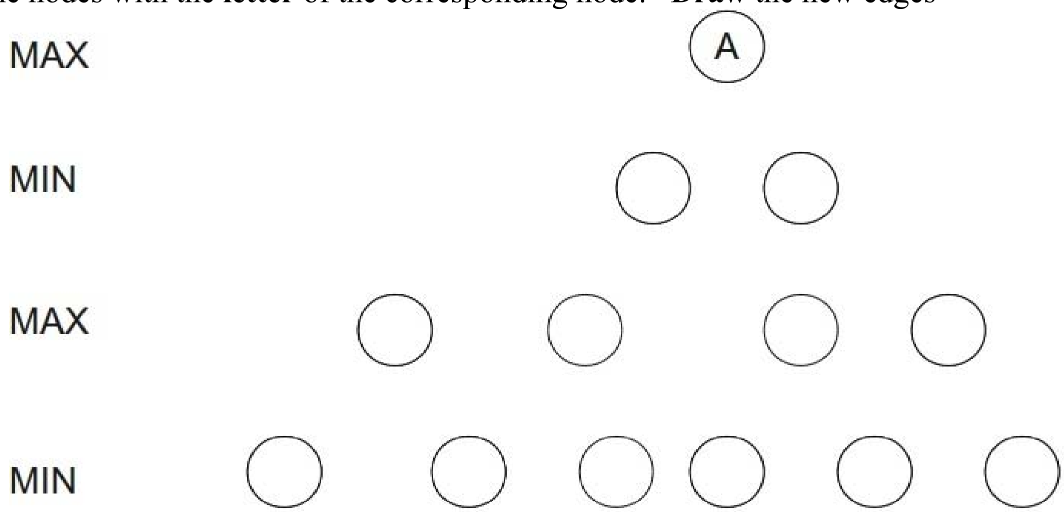
Perform Minimax on this tree. **Write the minimax value associated with each node in the box below, next to its corresponding node letter.**



A=	B=	C=	D=	E=	F=	G=

A2 Tree Rotations (8 points)

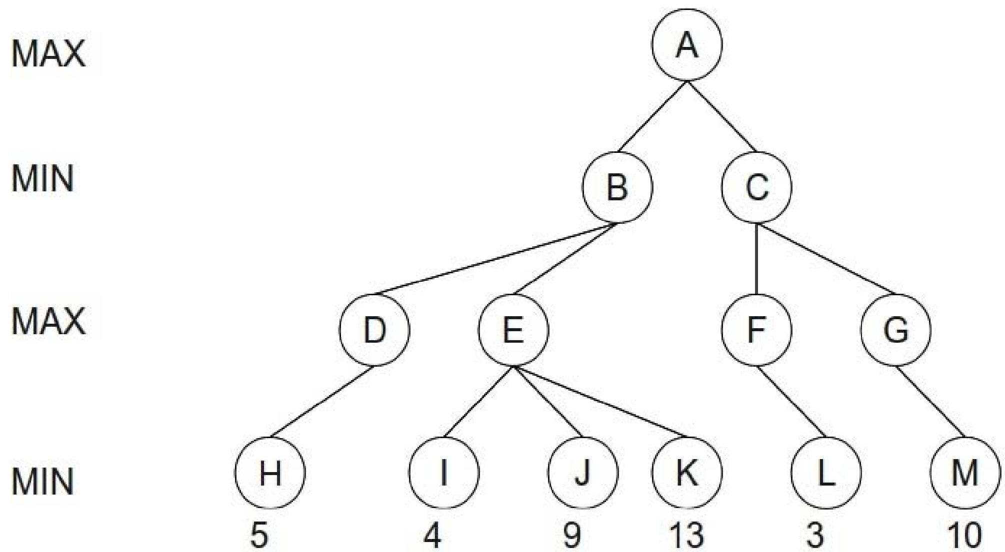
Using the minimax calculations from part A1, **without** performing any alpha-beta calculation, rotate the children of each node in the above tree at every level to ensure maximum alpha-beta pruning.
- **Fill in** the nodes with the **letter** of the corresponding node. **Draw** the new edges



Part B: Alpha Beta (35 points)

B1: Straight-forward Alpha Beta(15 points)

Perform Alpha Beta search on the following tree.
- **Indicate** pruning by striking through the appropriate edge(s).
- **Mark** your steps for partial credit.
- **Fill in** the number of static evaluations.
- **List** the leaf nodes **in the order** that they are statically evaluated.



Indicate in Next Move which of B or C you would go to from A and in Moving Towards which node in the bottom row you are heading toward.

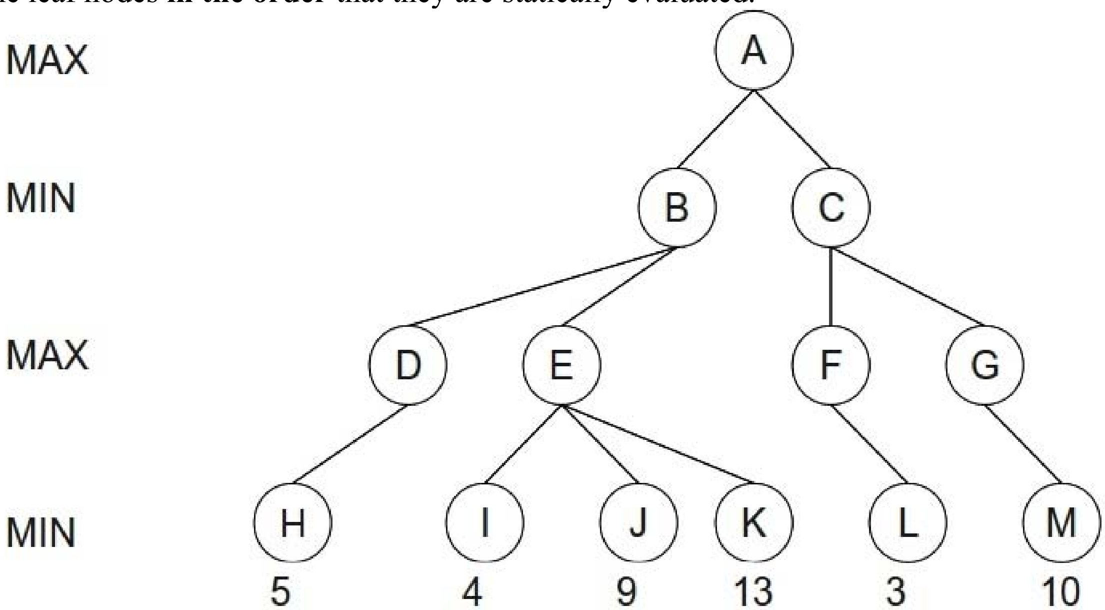
# of evaluations:______ List:____

Next Move: __________ Moving towards: _____________

B2: Preset Alpha-Beta (15 points)

Perform alpha-beta search, using **initial values** of **alpha = 5 and beta = 8**.

- **Indicate** pruning by striking through the appropriate edge(s).
- **Mark** your steps for partial credit.
- **Fill in** the number of static evaluations.
- **List** the leaf nodes **in the order** that they are statically evaluated.



Indicate in Next Move which of B or C you would go to from A and in Moving Towards which node in the bottom row you are heading toward.

# of evaluations:____ List:____

Next Move: __________ Moving towards: _____________

B3: Alpha-Beta Properties (5 points)

If you were able to maximally prune a tree while performing Alpha-Beta search, approaximately how many static evaluations would you end up doing for a tree of depth *d* and branching factor *b*?

Quiz 2, Problem 2,
Constraint Propagation (50 points)

After taking 6.034, your decide to offer your services to incoming freshman to help them set up their course schedules. One particular freshman comes to you with four classes as well as an availability schedule (grayed out boxes represent reserved times).

Course	Lecture Times Offered	Recitation Times Offered
3.091	MWF 11,12	TR 10,11,12,1
18.01	MWF 12, 1	TR 11,1,2
8.01T	MWF 10, 11, 12, 1, 2, 3	NONE
21F.301	MTWRF 10, 11	NONE

Time	MWF	TR
10		
11		
12		
1		
2		
3		

For easier bookkeeping you adopt the folowing naming convention (L = Lecture, R = Recitation, # = course number):

3.091 Lecture	→ **L3**	MWF10	→**10M**
3.091 Recitation	→ **R3**	MWF11	→**11M**
8.01T Lecture	→ **L8**		
18.01 Lecture	→ **L18**	TR10	→**10T**
18.01 Recitation	→ **R18**	TR11	→**11T**
21F.301 Lecture	→ **L21**		

You also devise this set of constraints for yourself:

1. Each class must be assigned to exactly one timeslot
2. Each timeslot can be assigned to a maximum of one class
3. No classes can be scheduled during the grayed out time periods
4. The TR selection for 21F.301 must occur at the same time as the MWF selection.

Part A: Picking a Representation (8 points)

In order to fill in this schedule, you decide to set it up as a CSP using meeting times as variables and courses as the values of in the domains. After filling in the domain table, this is what you see:

Variable	**Domain**
10M	L8, L21
11M	L3, L8, L21
12M	L3, L18, L8
1M	
2M	
3M	L8
10T	R3, L21
11T	
12T	R3
1T	
2T	R18
3T	

What is wrong with the way that this problem is set up and why?

Part B: Applying Constraints (42 points)

You decide to switch to a new representation that uses the courses as variables and the times as values.

B1 (5 points)

The initial domains are given below. Cross out the values that are incompatible with Constraint (3).

Variable	Domain					
L8	10M	11M	12M	1M	2M	3M
L3	11M	12M				
R3	10T	11T	12T	1T		
L21 (MWF)	10M	11M				
L21 (TR)	10T	11T				
L18	12M	1M				
R18	11T	1T	2T			

B2 (16 Points)

Run the DFS with **forward checking only** on your defined variables and the reduced domains you found in Part B1 by applying Constraint(3).

-

L8	
L3	
R3	
L21 (MWF)	
L21 (TR)	
L18	
R18	

B3 (5 Points)

How many times did the algorithm need to backtrack?

B4 (10 Points)

It occurs to you that you may be able to accelerate the process of finding a solution if you were to perform domain reduction with **propagation through singletons** *before* running the DFS. Fill in your updated domain table with the results of your computations.

Variable	**Domain**
L8	
L3	
R3	
L21 (MWF)	
L21 (TR)	
L18	
R18	

B5 (6 Points)

Run DFS with **constraint checking only** on your updated domain table:

●

L8	
L3	
R3	
L21 (MWF)	
L21 (TR)	
L18	
R18	

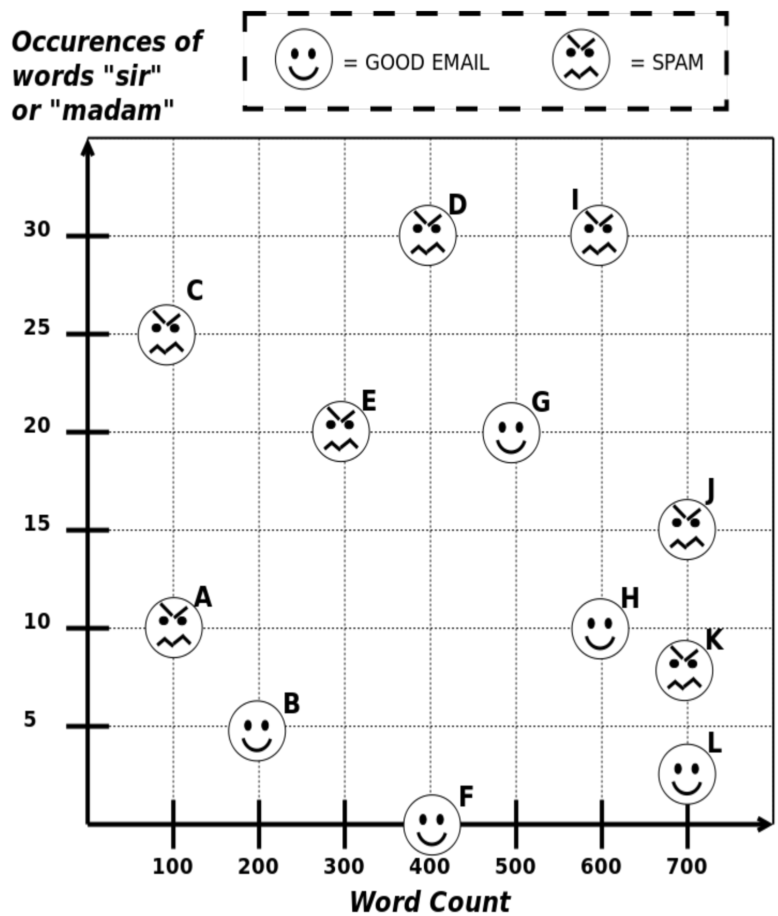
Quiz 3, Problem 1
KNN and ID Trees (50 points)

After receiving yet another "Dear sir or madam.." email, you decide to construct a spam filter.

Part A: Nearest Neighbors (25 points)

For your first attempt you decide to try using a k Nearest Neighbors model. You decide to classify spam according to 2 features: email word count and occurrences of the words "sir" or "madam".

A1 (10 points)



Draw the decision boundary for **1-nearest-neighbor** on the above diagram of the given training data. **Use the center of the faces as the positions of the training data points.**

A2 (8 points)

How will **1-nearest-neighbor** classify an email with **200** words of which **9** are the word “sir”?
Plot this point on the graph as X? (2pts)

How will **3-nearest-neighbors** classify an email with **600** words of which **7** are the word “madam”?
Plot this point on the graph as Y? (3pts)

How will **5-nearest-neighbors** classify an email with **500** words of which **25** are the word “madam”?
Plot this on the graph as Z? (3pts)

A3 (7 points)

List which points yield errors when performing leave-one-out cross validation using **1-nearest-neighbor** classification. (3 pts)

How would one go about selecting a good **k** to use? (4 pts)

Part B: ID Trees (25 points)

Realizing nearest neighbors may not be the best tool for building a spam filter, you decide to try another classifier you learned about in 6.034: Identification Trees.

B1 (8 points)

It appears that the over-use of the words “sir or madam” seems to be a strong hint at an email being spam.

What is the minimum disorder and minimum-disorder decision boundary when you consider only the dimension of “Sir or Madam”? You can use fractions, real numbers, and logarithms in your answer.

Approximate boundary:
Associated Disorder:

B2 (8 points)

Suppose we were given the following additional information about our training set:

Emails I, G, J, H, K and L are from Anne Hunter.

One of those emails might be important so you don't want to risk missing a single one so you re-label all Anne Hunter emails in the training data set to be good emails. You are to find the best axis-parallel test given the revised labellings of good email and spam.

(NOTE: Use the unlabeled graphs in the tear-off sheets section if you need it to visualize the modified data).

B2.1 Which emails does your new test missclassify on the **modified** data? **(4pts)**

B2.2 What is the disorder of your new test on the **modified** training data set?
Leave your answer as a function of fractions, real numbers, and logarithms. **(4pts)**

B3 (9 points)

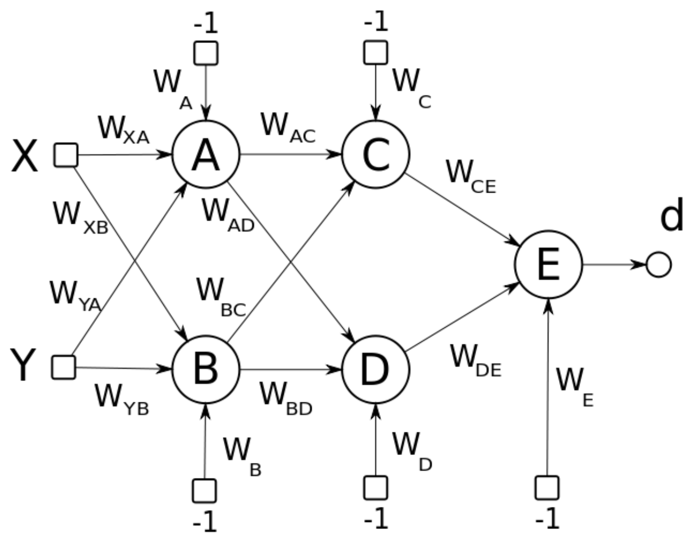
Soon, you decide that your life goal is no longer to be a tool for a Harvard or Sloanie startup so you decide that **all emails from Anne Hunter should be marked as spam. (Again, use the unlabeled graphs in the tear-off sheets if you need them).**

Given the revised determination of what is good email and spam, draw the **disorder minimizing identification tree** that represents your fully trained ID-tree spam filter. You may use any horizontal and vertical classifiers in the dimensions of word count and "sir or madam" occurrences. Ties should be broken in order of horizontal then vertical classifiers.

Quiz 3, Problem 2, Neural Nets (50 Points)

Note that this problem has three completely independent parts.

Part A: Derivations (14 pts)



A1. (7 pts) Using what you've learned from doing lab 5, write out the equation for $\frac{dP}{dw_{CE}}$ expressed in terms of o_i **,** d, and/or any weights and constants in the network. (o_i refers to the output of any neuron in the network.)

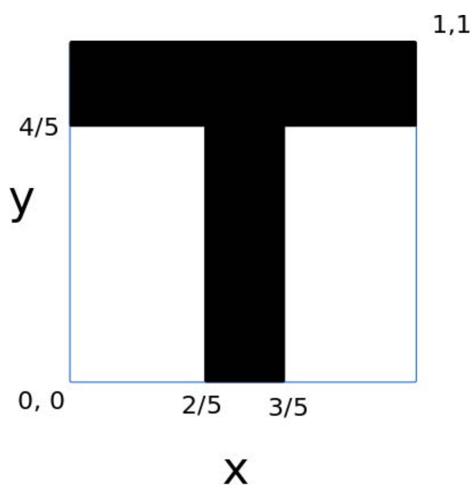
A2. (7 pts) Write out the equation for $\frac{do_E}{dw_{XA}}$ expressed in terms of $\frac{do_i}{dw_{XA}}$ **,** o_i**,** and/or any weights and constants in the network.

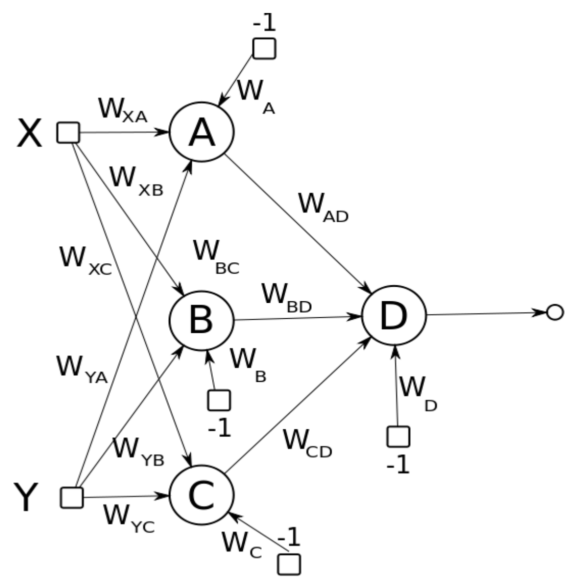
Part B: Letter Recognition (20 pts)

You propose to use a neural network to recognize characters from a scanned page. Letters are represented binary images on a 1x1 unit grid. Assume that scaling and rotation are all done.

Because you want to start with something easy, you start with the problem of recognizing a character as either possibly a T or definitely not a T. During training, each training sample consists of a random point, (x, y), along with the desired 0 or 1 value: 1 if the underlying pixel at (x, y) is part of a T; 0 if the pixel is part of a T's background.

You want to find the most **compact** network that will correctly handle the T problem, so you decide to analytically work out the minimal network that will correctly classify a character as possibly a T or definitely not a T.





Assume you decide to have the above network architecture, fill in the 7 missing weights in the table that are required to accurately classify all points in the image for T. Your weights must be **integer** weights or **integer** constraints on weights! Show your work for partial credit:

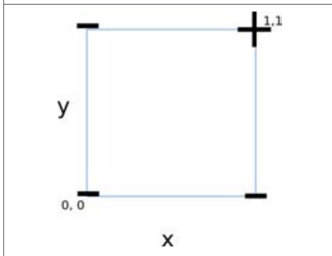
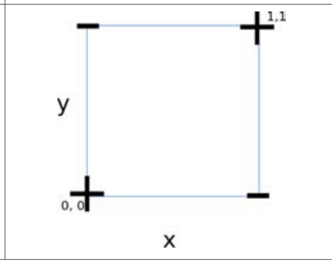
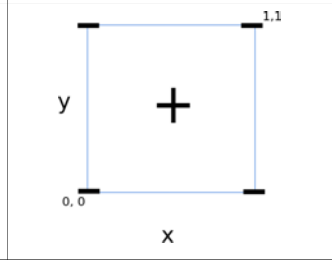
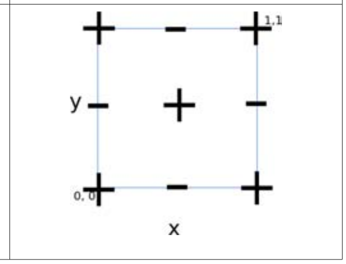
Answers:

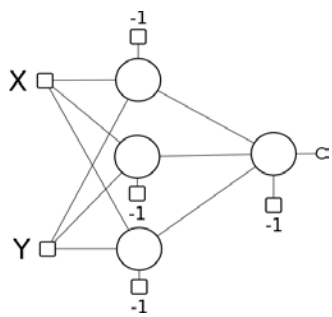
W_{XA}	**0**	**W_{XC}**	
W_{YA}		W_{YC}	**0**
W_{A}		**W_{C}**	
W_{XB}		**W_{AD}**	
W_{YB}		W_{BD}	**2**
W_{B}	**2**	W_{CD}	**2**
		W_{D}	**3**

Show work here for partial credit:

Part C: Expressive Power of Neural Nets (16 pts)

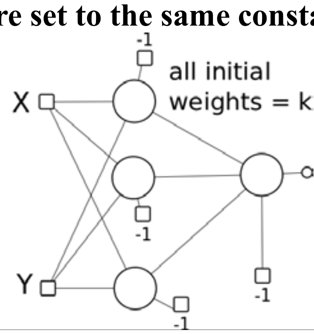
Circle all the functions that these networks are theoretically able to fully learn, and if your answer is No indicate the lowest possible error rate. **List of Functions:**

X AND Y	X = Y	X = 0.5 AND Y = 0.5	X-Shape
			



C1	Can be fully learned?	The minimum # error if No
X AND Y	Yes No	
X = Y	Yes No	
X = 0.5 AND Y = 0.5	Yes No	
X Shape	Yes No	

How about when all initial weights are set to the same constant k?

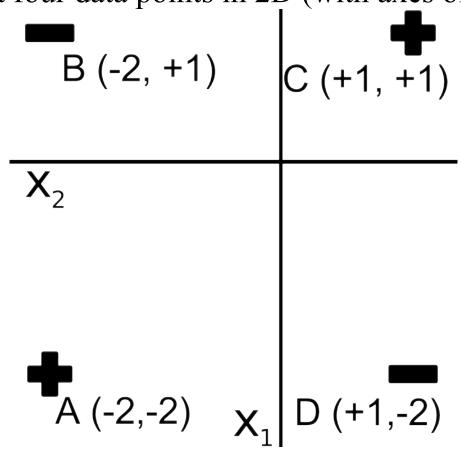


C2	Can be fully learned?	The minimum # error if No
X AND Y	Yes No	
X = Y	Yes No	
X = 0.5 AND Y = 0.5	Yes No	
X Shape	Yes No	

Quiz 4, Problem 1,
Support Vector Machines (50 points)

Part A: Solving SVMs (40 pts)

You decided to manually work out the SVM solution to the CORRELATES-WITH function. First you reduce your problem to looking at four data points in 2D (with axes of x_1 and x_2).



You ultimately want to come up with a classifier of the form below. **More formulas are provided in tear off sheets.**

$$h(\bar{x}): \sum a_i y_i K(\bar{x}, \bar{x}_i) + b \geq 0$$

Your TA, Yuan suggests that you use this kernel:

$$K(\vec{u}, \vec{v}) = u_1 v_1 + u_2 v_2 + |u_2 - u_1||v_2 - v_1|$$

A1(5 pts): Note that $\Phi(\vec{u})$ is a vector that is a transform of $\vec{u}$ and the kernel is the dot product, $\Phi(\vec{u}) \cdot \Phi(\vec{v})$. Determine the number of dimensions in $\Phi(\vec{u})$, and then write the components of $\Phi(\vec{u})$ in terms of $\vec{u}$'s x_1 and x_2 components, u_1 and u_2. Explain why it is better to use $\Phi(\vec{u})$ rather than $\vec{u}$.

$\Phi(\vec{u})$'s components:

Why better:

A2(10 pts): Fill in the unshaded portion in the following table of Kernel values.

K(A,A) =	K(A,B) =	K(A,C) =	K(A,D) =
K(B,A) =	K(B,B) =	K(B,C) =	K(B, D) =
K(C,A) =	K(C,B) =	K(C,C) =	K(C, D) =
K(D,A) =	K(D,B) =	K(D,C) =	K(D, D) =

A3 (10 pts): Now write out the full **constraints equations** you'll need to solve this SVM problem They should be in terms of α_i , b, and constants. **(Hint:** The four data points lie on their appropriate gutters).

	Constraint Equations
1	
2	
3	
4	
5	

A4 (5 pts): Instead of solving the system of equations for alphas, suppose the alphas were magically given to you as:

α_A=1/9	α_B=1/9	α_C=1/9	α_D=1/9	b=1

Compute $\vec{w}$ Given your alphas. **Hint: The number of dimensions in $\vec{w}$ is the same as the number of dimensions of $\Phi(\vec{x})$**

$\vec{w}$ =

A5 (5 pts): What is the equation of the optimal SVM decision boundary using the answers and values from A5?

$h(\vec{x}) \;=$

A6 (5 pts): What is the width of the road defined by the optimal SVM decision boundary?

Width =

Part B: Kernel for k-Nearest Neighbors (10 pts)

A student noticed that Kernel methods can be used in other classification methods, such as k-nearest neighbors. Specifically, one could classify an unknown point by summing up the fractional votes of all data points, each weighted using a Radial Basis Kernel. The final output of an unknown point $\vec{x}$ depends on the sum of **weighed** votes of data points $\vec{x}_i$ in the training set, computed using the function:

$$K(\vec{x}, \vec{x}_i) = \exp\left(\frac{-\|\vec{x} - \vec{x}_i\|}{s^2}\right)$$

Negatively labeled data points contribute -1 times the kernel value and positively labeled training data points contribute +1 times the kernel value.

You may find the graphs provided in the tear off sheets helpful.

B1 (6 pts)

Using this approach, as **s** increases to infinity, what k does this correspond to in k-NN?

As **s** decreases to zero, what k does this approximately correspond to in k-NN?

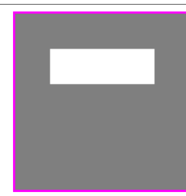
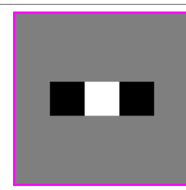
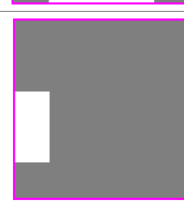
B2 (4 pts):

State a reason why you would prefer to use the SVM with Radial Basis Kernel solution rather than the method of Weighted Nearest Neighbors with a Gaussian Kernel.

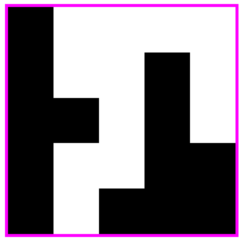
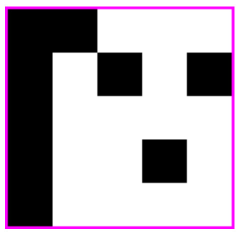
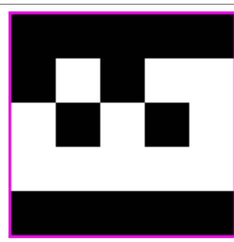
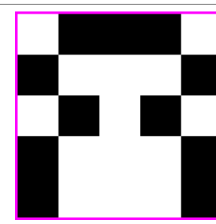
Quiz 4, Problem 2, Boosting (50 points)

Kenny wants to recognize faces in images. He comes up with a few things that he thinks will probably work well as weak classifiers and decides to create an amalgam classifier based on his training set. Then, given an image, he should be able to classify it as a FACE or NOT FACE. When matched against faces, the GRAY part of a classifier can be either WHITE or BLACK)

Classifiers:

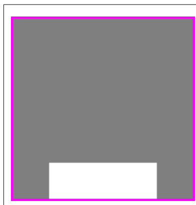
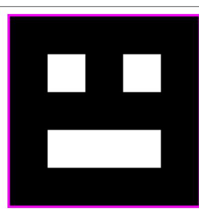
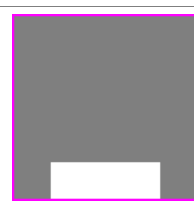
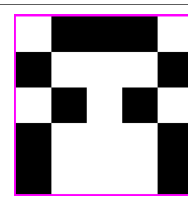
Name	Image Representation
A Has Hair	
B Has Forehead	
C Has Eyes	
D Has Nose	
E Has Smile	
F Has Ear	

Data:

Index	Classification	Image Representation		Index	Classification	Image Representation
1	NOT FACE			5	FACE	
2	NOT FACE			6	FACE	
3	NOT FACE			7	FACE	
4	FACE			8	FACE	

Examples:

Here is how classifier E works. Note that gray means "don't care," that is, it doesn't matter whether the pixel in the same position is black or white.

	+		= false			+		=	true

Part A: Pre-Boosting (10 points)

A1 Finding the Errors(5 points)

For each classifier, list the data points it gets wrong.

Classifier Name	General Idea	Misclassified
A	Has hair	2, 6, 7
B	Has forehead	4, 5
C	Has eyes	1, 5, 7
D	Has nose	
E	Has smile	3, 4, 7
F	Has ear	
G	TRUE (FACE)	
H	NOT(A)	
I	NOT(B)	
J	NOT(C)	
K	NOT(D)	
L	NOT(E)	
M	NOT(F)	
N	FALSE (NOT_FACE)	

A2 Weeding out Classifiers (5 points)

Which, if any, of the classifiers in A1 can never be useful and why?

Part B: Boosting (25 points)

B1 Synthesis (20 points)

Synthesize boosting using **only classifiers A, B, C, and E**. For ties, choose alphabetically.

	Round1		Round2		Round3		Round 4	
w1		h_1=		h_2=		h_3=		h_4=
w2		e_1=		e_2=		e_3=		e_4=
w3		a_1=		a_2=		a_3=		a_4=
w4								
w5								
w6								
w7								
w8								
e_A								
e_B								
e_C								
e_E								

B2 Amalgam Classifier (5 points)

What is the overall classifier after 4 rounds of boosting?

What is its error rate?

Part C: Miscellaneous (15 points)

C1 Using Classifiers (5 points)

Assume the boosting setup in Part B occurs for 100 rounds whether or not the overall classifier gets to the point where all training samples are correctly classified. Place the four classifiers in the following two bins

Frequently selected:	Infrequently selected:

C2 More using Classifiers (5 points)

Which **three** classifiers from A-N would you use so that you would not have to do boosting to get a perfect classifier for the samples.

C3 Even more using Classifiers (5 points)

Now, Suppose you are working with **just two classifiers,** neither of which has a zero error rate. Will boosting converge to an overall classifier that perfectly classifies all the training samples?

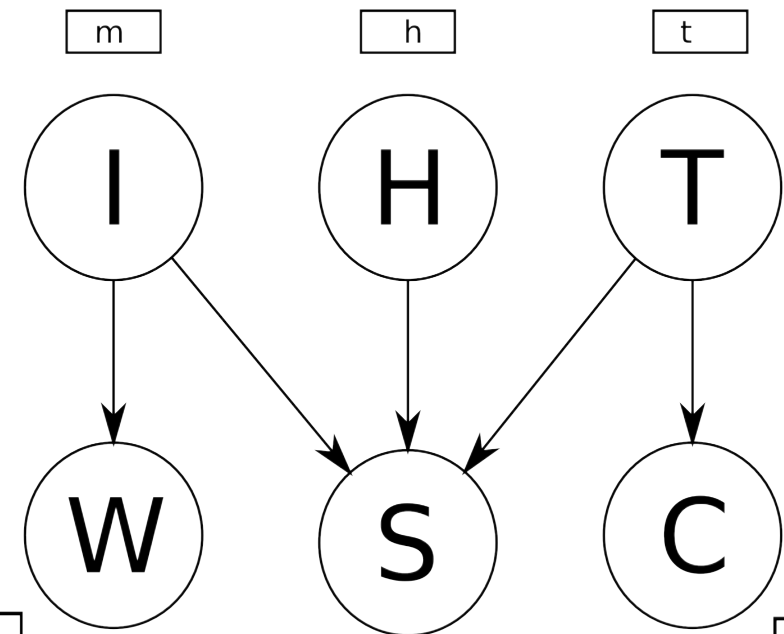
Yes No

Explain:

Quiz 5, Problem 1, Probability (50 points)

Part A: Life Lessons in Probability (22 pts)

Consider the following inference net developed for students who graduate from MIT:



I	W
F	v
T	w

I	H	T	S
F	F	F	a
F	F	T	b
F	T	F	c
F	T	T	d
T	F	F	e
T	F	T	f
T	T	F	g
T	T	T	s

T	C
F	j
T	k

I = Had quality **I**nstruction

H = **H**ard working ethic

T = Raw **T**alent

S = **S**uccessful in life

C = **C**onfidence

W = Took 6.034

A1: What is the probability that a 6.034 student (W = true) had quality instruction (I = true) and became successful in life (S = true), but did not have raw talent (T = false) yet was hardworking H = true) and confident (C = true). Leave your answer unsimplified in terms of constants from the probability tables. **(6pts)**

For A2-A3: Express your final answer in terms of expressions of probabilities that could be read off the Bayes Net. You do not need to simplify down to constants defined in the Bayes Net tables. You may use summations as necessary.

A2: What is probability of success in life (S = true) **given** that a student has high quality instruction (I = true)? **(6 pts)**

A3: What is the probability a student is hardworking (H = true) , **given** that s/he was a 6.034 student (W = true)? **(10 pts)**

Part B: The Naïve Crime Fighter (12 pts)

A murder has occurred in quiet town of South Park.   Your have been provided the following table by forensic experts:

Suspect	Has a motive	Location	Murder Weapon	Degree of Suspicion
Professor Chaos	0.4	Fair = 0.1 School = 0.7 CW= 0.2	Kindness = 0.3 Chili = 0.3 Moral Fiber = 0.2 Pure Evil = 0.2	0.3
Mint Berry Crunch	0.3	Fair = 0.4 School = 0.4 CW = 0.3	Kindness = 0.4 Chili = 0.1 Moral Fiber = 0.4 Pure Evil = 0.1	0.1
The Dark Lord Cthulu	0.1	Fair = 0.3 School = 0.3 CW = 0.4	Kindness = 0 Chili = 0.5 Moral Fiber = 0 Pure Evil = 0.5	0.4
Scott Tenorman	0.9	Fair = 0.8 School = 0.1 CW = 0.1	Kindness = 0.2 Chili = 0.5 Moral Fiber = 0.1 Pure Evil = 0.2	0.2

You have determined that the murder did not have a motive, the murder took place at the Fair, and the murder weapon was a bowl of chili.   You've decided to use what you learned about Naïve Bayes to help you determine who committed the murder.

The murderer is most likely:

Show your work:

Part C: Coin Tosses (16 pts)

You decide to reexamine the coin toss problem using model selection.

You have **2** types of coins:

1) **Fair:** P(heads) = 0.5
2) **All-heads:** P(heads) = 1.0

You have **2** possible models to describe your observed sequences of coin types and coin tosses.

1) **Model 1:** You have both types of coins and you draw one of them at random, with equal probability, and toss it exactly once. You repeat both the drawing and tossing 3 times total.
2) **Model 2:** You have both types of coins and you draw one of them at random, with equal probability, and toss it exactly 3 times.

Finally, you have the following observed data.

Toss 1	Toss 2	Toss 3
H	H	T

The following questions use your knowledge of model selection to determine which is most likely.

You decide to use the following criterion to weigh models:

$$P(M) = \frac{1}{Z}\frac{1}{|parameters|}$$ where Z is a normalization constant.

$|parameters|$ is defined as the number of cell entries in the CPTs of the Bayes Net representation.

C1. What is P(Model 1)? **(3 pts)** (Partial credit if you sketch the Models as Bayes Nets)

What is P(Model 2)? **(3 pts)**

You've decided that the a priori model probability P(Model) to use should be uniform.

P(Model 1) = P(Model 2)

Under this assumption you decide to work out the most likely model, given the data, P(Model | Data).

C2 What is the most-likely model based on this fully observed data set: **(10 pts)**

P(Model 1 | Data)?

P(Model 2 | Data)?

Therefore the most likely model is: (circle one) Model 1 Model 2

Quiz 5, Problem 2, Near Miss (20 points)

Having missed many tutorials, lectures, and recitations, Stew is stuck on trying to figure out who are the TAs in 6.034. You, who is more faithfully attending, knows who is who. Armed with your knowledge about Near Miss concept learning. You decide to build a model that will help Stew figure out who the TAs are.

The following table summarizes the training data about the current staff of 6.034. The **Title** attribute is organized as a tree, with MEng and PhDs both a type of Student. Students and Faculty are grouped under the type People-You-See-On-Campus

Name	TA	Hair Color	Title	Glasses	Gender	Experience # Years	Taken 6.034
Kendra	Yes	Brown	MEng	yes	female	1	Yes
Kenny	Yes	Brown	MEng	no	male	1	Yes
Martin	Yes	Black	MEng	no	male	1	Yes
Mark	Yes	Black	PhD	no	male	4	Yes
Mich	Yes	Blonde	MEng	yes	male	10	No
Gleb	Yes	Brown	MEng	no	male	2	Yes
Yuan	Yes	Black	PhD	yes	male	3	No
Lisa	No	Blond	Professor	yes	female	10	No
Bob	No	Brown	Professor	no	male	10	No
Randy	No	Brown	Professor	no	male	10	No

Fill in the table to build a model of a TA. Mark an attributes as "?" if the property has been dropped.

Example	Heuristics Used	Model Description TAs					
		Hair Color	Title	Glasses	Gender	Experience	Taken
Kendra	Initial Model	Brown	MEng	yes	female	1	Yes
Kenny							
Martin							
Yuan							
Bob							

Fill in the following table to build a model of a Faculty Member (FM).   You may assume that Patrick, Randy, Bob, and Lisa are faculty members.

Example	Heuristics Used	Model Description FMs					
		Hair Color	Title	Glasses	Gender	Experience	Taken
Patrick	Initial Model	Blond	Professor	Yes	Male	10	No
Randy							
Bob							
Mich							
Lisa							

What class(es) would match these people given your TA model and your FM model. If neither, write N in the Class(es) column.

Name	Class(es)	Hair Color	Title	Glasses	Gender	Experience	Taken 6.034
Olga		Blond	MEng	no	female	1	Yes
Patricia		Blond	Professor	Yes	female	10	No

Quiz 5, Problem 3, Big Ideas (30 points)

Circle the **best** answer for each of the following question. There is no penalty for wrong answers, so it pays to guess in the absence of knowledge.

1 Ullman's alignment method for object recognition

1. Is an effort to use neural nets to detect faces aligned with a specified orientation
2. Relies on a presumed ability to put visual features in correspondence
3. Uses A* to calculate the transform needed to synthesize a view
4. Uses a forward chaining rule set
5. None of the above

2 Ullman's intermediate-features method for object recognition

1. Is an effort to use boosting with a classifier count not too small and not too large
2. Is an example of the Rumpelstiltskin principle
3. Is a demonstration of the power of large data sets drawn from the internet
4. Uses libraries containing samples (such as nose and mouth combinations) to recognize faces
5. None of the above

3 The SOAR architecture is best described as

1. A commitment to the strong story hypothesis
2. A commitment to rule-like information processing
3. An effort to build systems with parts that fail
4. The design philosophy that led to the Python programming language
5. None of the above

4 The Genesis architecture (Winston's research focus) is best described as, in part, as

1. A commitment to the strong story hypothesis
2. Primarily motivated by a desire to build more intelligent commercial systems
3. A commitment to rule-like information processing
4. A belief that the human species became gradually smarter over 100s of thousands of years.
5. None of the above

5 A transition frame

1. Focuses on movement along a trajectory
2. Focuses on the movement from childlike to adult thinking
3. Focuses on a small vocabulary of state changes
4. Provides a mechanism for inheriting slots from abstract frames, such as the disaster frame
5. None of the above

6 Reification is

1. The attempt to develop a universal representation
2. The tendency to attribute magical powers to particular mechanisms
3. The process by which ways of thinking are determined by macro and micro cultures
4. The process of using perceptions to answer questions too hard for rule-based systems
5. None of the above

7 Arch learning includes

1. A demonstration of how to combine the benefits of neural nets and genetic algorithms
2. A commitment to bulldozer computing using 100's of examples to learn concepts
3. The near miss concept
4. A central role for the Goldilocks principle
5. None of the above

8 Arch learning benefits importantly from

1. An intelligent teacher
2. Exposure to all samples at the same time
3. Use of crossover
4. Sparse spaces
5. None of the above

9 Experimental evidence indicates

1. People who talk to themselves more are better at physics problems than those who talk less
2. Disoriented rats look for hidden food in random corners of a rectangular room
3. Disoriented children combine color and shape information at about the time they start walking
4. Disoriented children combine color and shape information at about the time they start counting
5. None of the above

10 Goal trees

1. Enable rule-based systems to avoid logical inconsistency
2. Enable rule-based systems answer questions about behavior
3. Are central to the subsumption architecture's ability to operate without environment models
4. Are central to the subsumption architecture's ability to cope with unreliable hardware
5. None of the above