CS 540:
Introduction to Artificial Intelligence

Final Exam: 1:00-3:30 pm, August 8, 2003
Room 265 Materials Sciences Building

CLOSED BOOK
(two-sided sheet of handwritten notes and a calculator allowed)

Write your answers on these pages and show your work. If you feel that a question is not fully specified, state any assumptions you need to make in order to solve the problem. You may use the backs of these sheets for scratch work. Budget your time wisely.

Before you begin, write your name on every page of the exam and read through all the questions (as some have multiple parts and are more involved than others). Make sure your exam contains ***seven (7)*** problems on ***eleven (11)*** pages.

Name **Solution** _____

Student ID **000-000-0000** _____

Problem	Score	Max Score
1	_____	20
2	_____	15
3	_____	25
4	_____	15
5	_____	30
6	_____	10
7	_____	10
TOTAL	_____	125

Name: _____

Problem 1: General Search Questions (20 points)

a) Answer each of the following questions *true* or *false*:

i. Tabu search with a horizon of 1 behaves the same as a greedy hill-climbing search.

***True* - hill-climbing shouldn't return to it's last state anyway**

ii. Simulated annealing with a temperature $T = 0$ also behaves identically to a greedy hill-climbing search.

***False* - SA can still take uphill but sub-optimal moves**

iii. Breadth-first search is always a complete search method, even if all of the actions have different costs.

***True* - completeness refers to finding "a" solution, not the "best"**

iv. When hill-climbing and greedy best first search use the exact same admissible heuristic function, they will expand the same set of search nodes.

***False* - greedy best-first can backtrack (keeps an open list)**

v. If two admissible heuristic functions evaluate the same search node n as $h_1(n) = 6$ and $h_2(n) = 8$, we say h_1 dominates h_2, because it is less likely to overestimate the actual cost.

***False* - it's the other way around (h_2 is closer to the actual cost)**

b) Provide a short answer to the following questions:

i. What are the three basic components of any genetic algorithm?

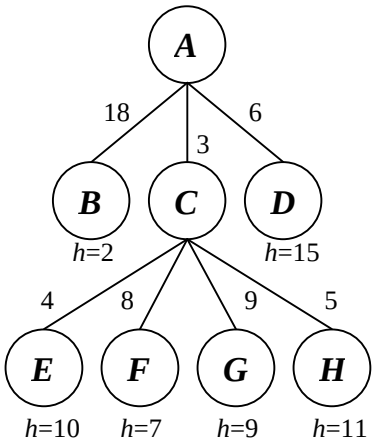
1. natural selection (fitness)
2. reproduction (crossover)
3. mutation

ii. Compare and contrast genetic algorithms to beam search.

Both maintain a fixed-size set of solutions which are the best (throwing out or killing the worst solutions). GAs maintain a population, whereas beam search keeps, well, a "beam." The main difference is in selecting neighbors, beam search hill-climbs, where beam search "reproduces" (the former is local, the latter is non-local).

Name: _____

c) Consider the following *partial search tree*, where edges are labeled with actual costs of the associated action, and each node is labeled with its heuristic evaluation. Which node will be expanded next by each of the following search methods?



i. Greedy best-first search: **B**

ii. Uniform cost search: **D**

iii. A* search: **E**

d) Your good friend is the groundskeeper at the mansion of mean old Mr. Mathis. In the backyard, there is a huge fountain with a complex network of pipes controlled by over 100 valves. One weekend, Mr. Mathis announces he's going on vacation, and when he returns he wants the fountain to spray as high as it can… but the plans for the pipe network have been lost! Plus, since she only has the weekend, your friend can't possibly try *all* of the valve combinations to find the optimal setting. Which local optimization search method might she want to use *in real life* (since the fountain can't be simulated on a computer) to maximize the height of the fountain? You may assume that a valve is either *on* or *off*, and the water height is easily measured. If you need to make any other assumptions, state them clearly.

There are several possibilities. Perhaps the most *plausible* is simulated annealing: your friend would only have to choose one valve at random and see if it helped. If it did, she'd just try another valve. If it didn't make the fountain higher, she might leave it and she might not, etc. (presumably you know how SA works) Tabu search is also plausible, but still requires trying *every* valve (more than 100) the first time. Hill-climbing is even worse and a genetic algorithm would be downright impossible to do over a weekend!

Name: _____

Problem 2: Logic and Planning Questions (15 points)

e) Provide a short answer to each of the following questions:

i. Are the literals $P(F(y), y, x)$ and $P(x, F(A), F(v))$ unifiable? If so, show the unifying substitution θ. If not, explain why.

Yes. $\theta = \{x/F(y), y/F(A), y/v\}$

ii. Convert this FOL sentence to conjunctive normal form: $\forall x \exists y\ Owns(x,y) \Rightarrow \neg Like(x,y)$

1. replace implication $\forall x \exists y\ \neg Owns(x,y) \vee \neg Like(x,y)$
2. Skolemeize *y* $\forall x\ \neg Owns(x, F(x)) \vee \neg Like(x, F(x))$
3. Drop universals $\neg Owns(x, F(x)) \vee \neg Like(x, F(x))$

iii. What is the frame problem in situation calculus? How do we deal with it?

The frame problem is when we have axioms that describe what changes in the world as a result of an action, but nothing to describe what does *not* change. This can be fixed by writing *frame axioms* or by using a closed world assumption: nothing changes unless explicitly described (equivalent to a lot of picky frame axioms)

f) Use forward chaining to show that $Bigger(Smaug, Bilbo)$ is entailed by the following KB. Indicate which sentences were used with GMP, and show each substitution θ.

1. $Hobbit(Bilbo)$
2. $Wizard(Gandalf)$
3. $Dragon(Smaug)$
4. $Dragon(x) \wedge Wizard(y) \Rightarrow Bigger(x,y)$
5. $Wizard(x) \wedge Hobbit(y) \Rightarrow Bigger(x,y)$
6. $Bigger(x,y) \wedge Bigger(y,z) \Rightarrow Bigger(x,z)$

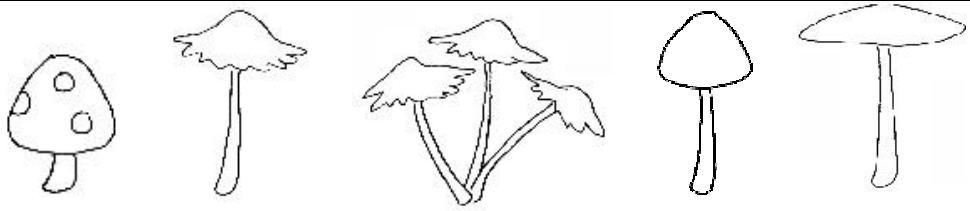
7. $Bigger(Smaug, Gandalf)$ **(2,3,4: $\theta = \{x/Smaug, y/Gandalf\}$)**
8. $Bigger(Gandalf, Bilbo)$ **(1,3,5: $\theta = \{x/Gandalf, y/Bilbo\}$)**
9. $Bigger(Smaug, Bilbo)$ **(6,7,8: $\theta = \{x/Smaug, y/Gandalf, z/Bilbo\}$)**

Name: _____

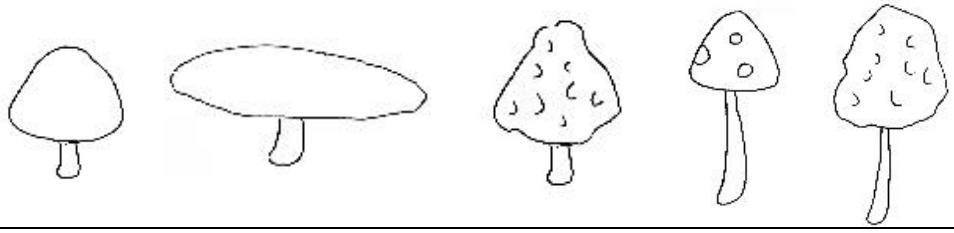
Problem 3: General Learning Algorithms (25 points)

Consider the task of learning to identify mushrooms that are SAFE or POISONOUS to eat based on a set of physical features. Four Boolean and discrete valued features that you could use are: STEM = {short, long}, BELL = {rounded, flat}, TEXTURE = {plain, spots, bumpy, ruffles}, and NUMBER = {single, multiple}. Consider using these features on the following training data:

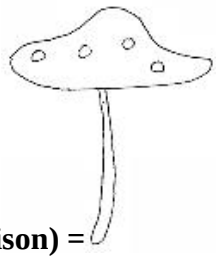
SAFE:



POISONOUS:



a) How would the naïve Bayes classifier label the following example:



$P(\text{safe}) \times P(\text{long} \mid \text{safe}) \times P(\text{flat} \mid \text{safe}) \times P(\text{spots} \mid \text{safe}) \times P(\text{single} \mid \text{safe}) =$
$(.5)(.8)(.6)(.2)(.8) = 0.0384$

$P(\text{poison}) \times P(\text{long} \mid \text{poison}) \times P(\text{flat} \mid \text{poison}) \times P(\text{spots} \mid \text{poison}) \times P(\text{single} \mid \text{poison}) =$
$(.5)(.4)(.2)(.2)(1.0) = 0.008$

SAFE is the more probable classification

b) How would 3-nearest neighbors, using hamming distance and unweighted voting, classify the same example from part (a)?

The 3 nearest neighbors (each with distance 1) are:

The vote is 2-3 in favor of SAFE

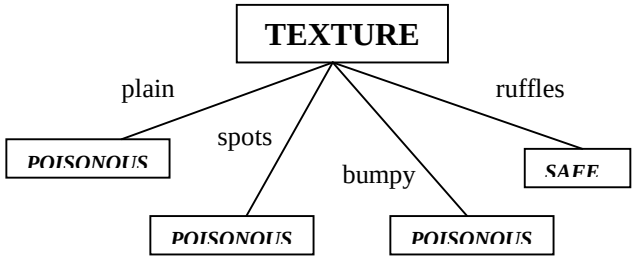
Name: _____

c) Use information gain to choose between TEXTURE and NUMBER to use in a *decision stump* (a decision tree with only one internal node at the root). Draw a diagram of the learned stump, and break ties at classification nodes by labeling them as *POISONOUS* (just to be on the safe side). Show all your work for partial credit.

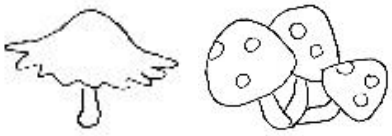
InfoGain(TEXTURE)
= Entropy(S) – (.4)Entropy(S_{plain}) – (.2)Entropy(S_{spots}) – (.2)Entropy(S_{bumpy}) – (.2)Entropy($S_{ruffles}$)
= 1 – (.4)(0) – (.2)(0) – (.2)(1) – (.2)(1)
= 0.4

InfoGain(NUMBER)
= Entropy(S) – (.9)Entropy(S_{single}) – (.1)Entropy($S_{multiple}$)
= 1 – (.9)(.991) – (.1)(1)
= 0.108

TEXTURE is the winner:

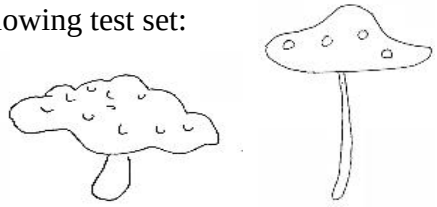


d) Estimate the future accuracy of this decision stump given the following test set:



← ***SAFE***

POISONOUS →



75% accuracy (misclassifies the second safe example)

e) Draw a perceptron that might be used to learn this problem (just the structure is fine, you do not need to label with weights).

There should be 7 inputs and 1 threshold. Since STEM, BELL, and NUMBER are all Boolean, we can use one Boolean input for each of them (3). But TEXTURE is discrete, so a separate Boolean input should be generated for each possible value (4).

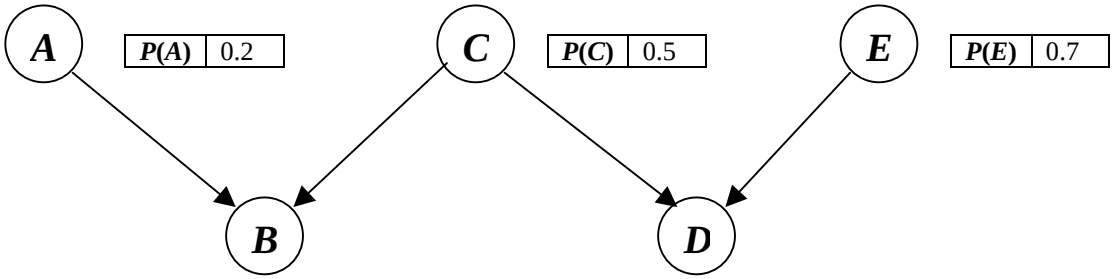
f) What is the dimensionality of a perceptron's hypothesis space for this problem?

The hypothesis is *weight space*, and has dimentionality of 8 (7 inputs + 1 threshold)

Name: _____

Problem 4: Bayesian Networks (15 points)

Consider the following Bayesian network where all variables are Boolean:



	A,C	*¬A,C*	*A,¬C*	*¬A,¬C*
***P*(*B*)**	0.9	0.6	0.4	0.5

	C,E	*¬C,E*	*C,¬E*	*¬C,¬E*
***P*(*D*)**	0.3	0.1	0.8	0.7

a) How large would the full-joint probability table (FJPT) have to be for this problem?

$2^5 = 32$

b) What is the joint probability that all five variables are simultaneously true?

$$P(A,B,C,D,E)$$
$$= P(A) \times P(B \mid A,C) \times P(C) \times P(D \mid C,E) \times P(E)$$
$$= (.2)(.9)(.5)(.3)(.7)$$
$$= 0.0189$$

c) Compute the probability $P(\neg C \mid A, \neg B, D, \neg E)$

$$P(\neg C \mid A, \neg B, D, \neg E) = P(A, \neg B, \neg C, D, \neg E) \,/\, P(A, \neg B, \neg C, D, \neg E) + P(A, \neg B, C, D, \neg E)$$

$$P(A, \neg B, \neg C, D, \neg E) = P(A) \times P(\neg B \mid A, \neg C) \times P(\neg C) \times P(D \mid \neg C, \neg E) \times P(\neg E)$$
$$= (.2)(.6)(.5)(.7)(.3) = 0.0126$$

$$P(A, \neg B, C, D, \neg E) = P(A) \times P(\neg B \mid A, C) \times P(C) \times P(D \mid C, \neg E) \times P(\neg E)$$
$$= (.2)(.1)(.5)(.8)(.3) = 0.0024$$

Thus, $P(\neg C \mid A, \neg B, D, \neg E) = 0.0126 / (0.0126 + 0.0024) = 0.84$

Name: _____

Problem 5: Machine Learning Questions (30 points)

Provide a short answer to each of the following questions:

a) What's the difference between *eager* and *lazy* learning algortihms? Give an example of each.

Lazy: memorized examples and compares at test time (e.g. k-nearest neighbors). Eager: actively constructs a model hypothesis function (e.g. ID3)

b) Give an example of a *regression* learning task. Which of the machine learning algorithms we've discussed can learn this kind of concept well?

Regression tasks learn real-valued (continuous) functions. Such as predicting the exact temperature tomorrow based on a set of recent meteorological data.

c) Give one advantage and one disadvantage of Bayesian belief networks over naïve Bayes.

Advantages: may use conditional independence, can deal with hidden variables, can query any variable in the network

Disadvantages: usually more space complex, slower to train or perform inference

d) Would the hypotheses learned by an inductive logic programming (ILP) system be considered *symbolic* or *connectionist* AI? Breifly explain.

Definitely symbolic. Recall that in symbolic AI each part of the agent is imbued with specific meaning. The predictes ILP uses to create theories has such meaning. In connectionist AI, such meaning may exist, but it emerges from all the parts of the egent being interconnected, and we can't readily comprehend them (e.g. neural networks)

Name: _____

e) What is the *curse of dimensionality*? Which learning methods are most affected by it? Describe one approach to dealing with this problem.

The curse of dimensionality is the problem of having many features, many of which are irrelevant, and the learning algorithm has a hard time modeling the function in such a high-dimensional space. *k*-NN and naïve Bayes are probably the most affected by it… and it can often be dealt with by using feature selection (ranked feature selection with info-gain for example, or forward/backward chaining)

f) What relationship does the *minimum description length* principle have with the ID3 learning algorithm? What about backpropagation learning for neural networks?

The MDL principle illustrates the tradeoff between a hypothesis' complexity (size) and performance (# misclassified examples). Since any complete decision tree with fit the data the "performance" on training data is constant, thus ID3 uses information gain as a heuristic to find the smallest (least complex) tree. With backpropagation, however, the complexity is constant (the number fo weights doesn't change), so it tries to optimize perfromance using gradient descent.

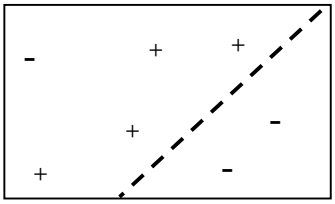
g) Ensemble learning algorithms often benefit from the combined knowledge of its constituent agents, which learn from *slightly* different subsets of examples. Imagine that we want to create and ensemble learner where each constituent agent learns from subsets that are *very* different. How might we do this? How might the ensemble then classify new examples? Clearly state any assumptions that you need to.

The thing that perhaps we should do is this: choose how many classifiers we want… let's say *k* classifiers. The use k-means clustering to partition the data into k very distinct subsets, and train a different classifier one each cluster. At test time, decide which cluster this particular test example belongs to, and let the corresponding classifier label it (or perhaps perform a weighted vote based on the distance of the example to each cluster center, etc…)

Name: _____

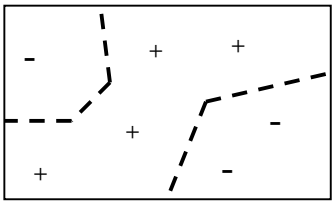
Problem 6: Feature Space (10 points)

The following Venn diagrams represent training examples for a Boolean-valued concept function plotted in *feature space*. Show how each of the following machine learning algorithms *might* partition the space based on these examples. Briefly explain to the right of each diagram why that algorithm would partition the data that particular way. (No need for calculations… just give a qualitative answer.)



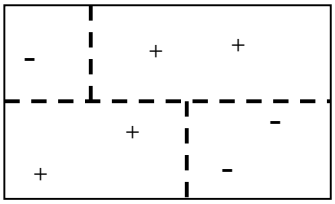
Naïve Bayes

Learns a single linear hyperplane to separate the data



1-Nearest Neighbor

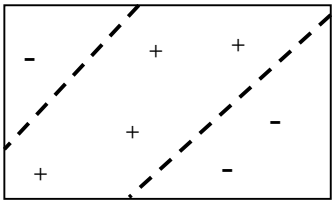
Construct a convex polygon around each example, also known as a "Varnoi diagram"



ID3

Recursively draws axis-parallel separating planes to subdivide the data

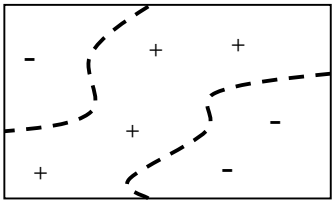
---



Neural Network with 1 hidden layer of 2 units

May learn exactly two separating hyperplanes (one from each hidden unit) and con/dis-join them

---



Neural Network with unlimited hidden layers/units

Can learn an arbitrarily complex or expressive separation of the data

Name: _____

Problem 7: Miscellaneous Definitions (10 points)

Briefly define the following concepts and explain the significance of each to A.I.
(Write your answer *below* the phrase.)

α-β Search

Variant of MiniMax search in game-playing that uses α-β pruning. The pruning allows the agent to disregard branches of the game tree that are fruitless (heh-heh) and ultimately search deeper, making a theoretically better game-playing agent.

Occam's Razor

All things being equal, choose the simplest solution. This in machine learning this means choosing the simplest hypothesis from among those that *perform equally well*.

Tuning Sets

A held-aside subset of the training data used to "tune" the hypothesis function to reduce overfitting on the training data. ID3 can use tuning sets to "prune" nodes, *k*-NN can use tuning sets to choose the best *k*, and backpropagation can use them to decide when to stop training.

Promotion / Demotion

In partial order planners, some actions can "clobber" or "threat" others: the effects of a particular action cancel out the preconditions for another. In this case, POP algorithms "promote" (or "demote") actions to remove such conflicts.

The Markov Assumption

When analyzing sequence data, even observation is only dependent on a limited series of observations before it, not the entire sequence. This is used in Markov chain models (MCMs) and hidden Markov models (HMMs) for sequence classification and recognition.