CS533 — Written Homework 3 — Solutions

1. Consider the following training examples. Compute the mutual information between each of the features f1, f2, f3, and f4, and the class. What feature will C4.5 choose for the root of the decision tree using the mutual information heuristic?

class	f1	f2	f3	f4
c1	0	0	1	0
c1	0	0	0	1
c1	0	1	0	0
c2	0	0	0	1
c2	0	1	0	0
c2	0	1	0	1
c2	1	0	0	1

The following table shows the distribution of examples from c1 and c2 that we would get under each of the splits and the computed mutual information:

split	left child	right child	mutual information
f1	3 3	0 1	.1281
f2	2 2	1 2	.0202
f3	2 4	1 0	.1981
f4	2 1	1 3	.1281

C4.5 will chose to split on f3.

2. A learning problem that is very hard for learning algorithms is the parity problem. In parity, the class is c2 if an odd number of the features are 1, else the class is c1. For example, here are all 8 possible examples for 3-bit parity:

class	f1	f2	f3
c1	0	0	0
c2	0	0	1
c2	0	1	0
c1	0	1	1
c2	1	0	0
c1	1	0	1
c1	1	1	0
c2	1	1	1

If C4.5 were given these examples, what would it compute as the mutual information for each feature? Show an example of a decision tree that correctly classifies all examples of 3-parity.

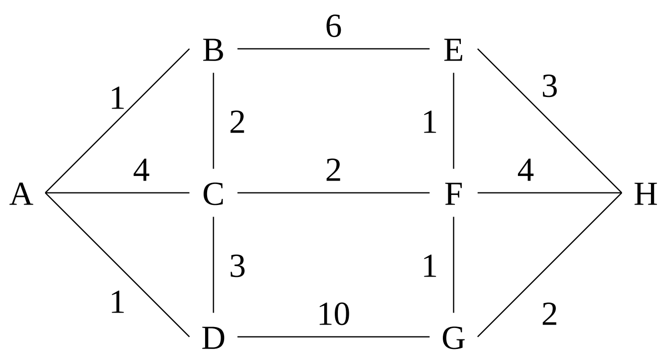
The mutual information is 0 for each feature, so C4.5 will choose a split at random.

Here is one possible tree in indented format

```
f1=0:
  f2=0:
    f3=0: c1
    f3=1: c2
  f2=1:
    f3=0: c2
    f3=1: c1
f1=1:
  f2=0:
    f3=0: c2
    f3=1: c1
  f2=1:
    f3=0: c1
    f3=1: c2
```

A complete binary tree of depth 3 is required.

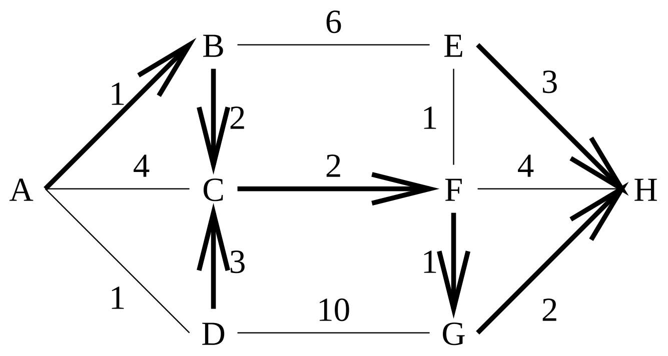
3. Compute the optimal value function and the optimal policy for the following graph problem like the one on the slides in class. The start state is A and the goal state is H. Each path connecting two states is labeled with its cost. Assume that the initial value function is Infinite at every node. What is the minimum number of "backups" required to compute this value function? List the states in order of when they should be backed up. What would happen if the value function was initialized to 0 at every node?



The value function is as follows:

A	B	C	D	E	F	G	H
8	7	5	8	3	3	2	0

The policy is shown by heavy arrows on this figure:



The minimum number of backups is 7: one for every state except the goal state. We must do these backups in reverse order along the optimal policy in the order G, E, F, C, D, B, A. In the following table I denotes positive infinity, and each line shows the results of one backup.

	A	B	C	D	E	F	G	H
init	I	I	I	I	I	I	I	0
G	I	I	I	I	I	I	2	0
E	I	I	I	I	3	I	2	0
F	I	I	I	I	3	3	2	0
C	I	I	5	I	3	3	2	0
D	I	I	5	8	3	3	2	0
B	I	7	5	8	3	3	2	0
A	8	7	5	8	3	3	2	0

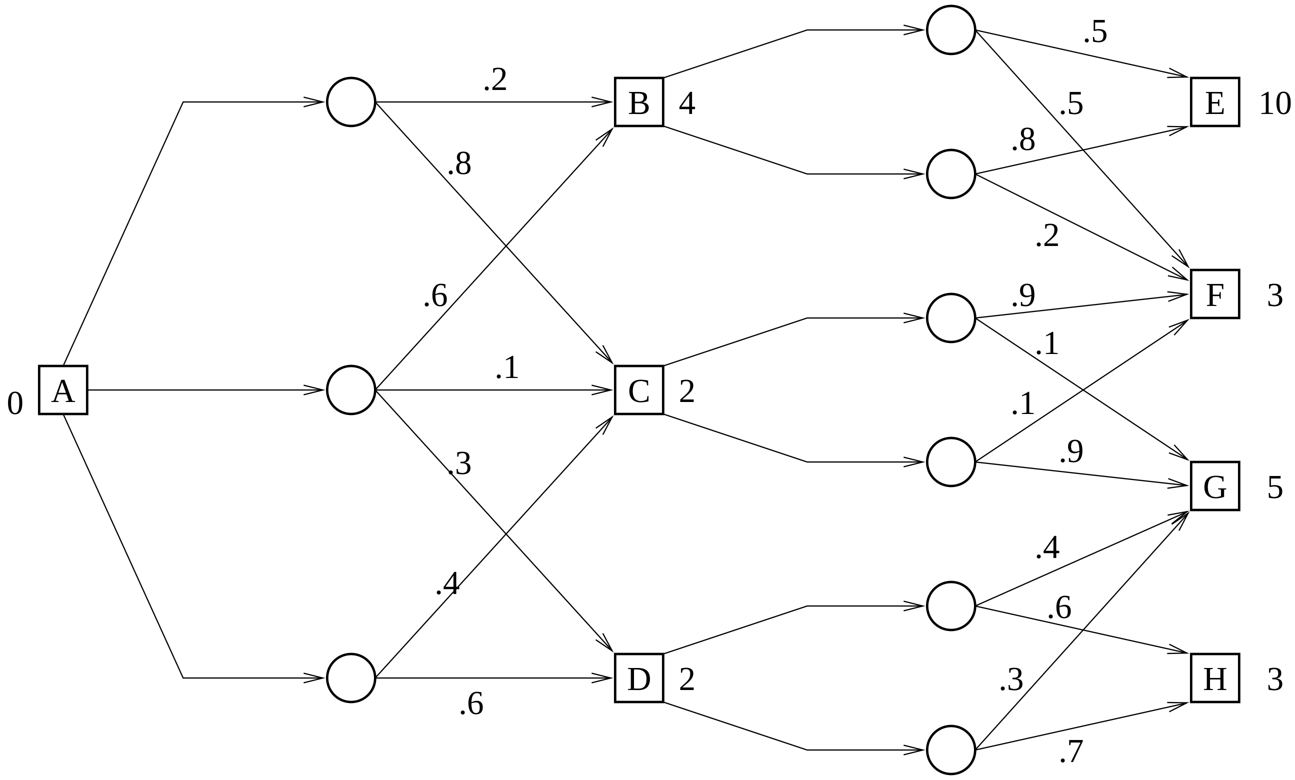
If the values are initialized to zero, more backups are required. An asterisk indicates the "final" backup for each state (i.e., the backup that produced the correct value).

	A	B	C	D	E	F	G	H	values computed
init	0	0	0	0	0	0	0	0	
G	0	0	0	0	0	0	1	0	F:1 H:2 D:10
F	0	0	0	0	0	1	1	0	E:1 H:4 G: 2 C:2
G*	0	0	0	0	0	1	2	0	F:2 H:2 D:10
E	0	0	0	0	2	1	2	0	F:2 H:3 B:6
F	0	0	0	0	2	2	2	0	C:2 G:3 H:4 E:3
E*	0	0	0	0	3	2	2	0	H:3 F:3 B:6
C	0	0	2	0	3	2	2	0	B:2 D:3 A:4 F:5
F*	0	0	2	0	3	3	2	0	G:3 C:4 H:4 E:4
B	0	1	2	0	3	3	2	0	A:1 C:4 A:4 E:9
D	0	1	2	1	3	3	2	0	A:1 C:5 G:12
A	2	1	2	1	3	3	2	0	B:2 D:2 C:6
C	2	1	4	1	3	3	2	0	B:4 D:4 F:5 A:6
B	2	3	4	1	3	3	2	0	A:3 C:6 E:9
D	2	3	4	3	3	3	2	0	A:3 C:7 G:12
A	4	3	4	3	3	3	2	0	B:4 D:4 C:8
C*	4	3	5	3	3	3	2	0	B:5 F:5 D:6 A:8
B	4	5	5	3	3	3	2	0	A:5 C:7 E:9
D	4	5	5	5	3	3	2	0	A:5 C:8 G:12
A	6	5	5	5	3	3	2	0	B:6 D:6 C:9
B*	6	7	5	5	3	3	2	0	A:7 C:7 E:9
D	6	7	5	7	3	3	2	0	A:7 C:8 G:12
A*	8	7	5	7	3	3	2	0	B:8 C:9 D:8
D	8	7	5	8	3	3	2	0	C:8 A:9 G:12

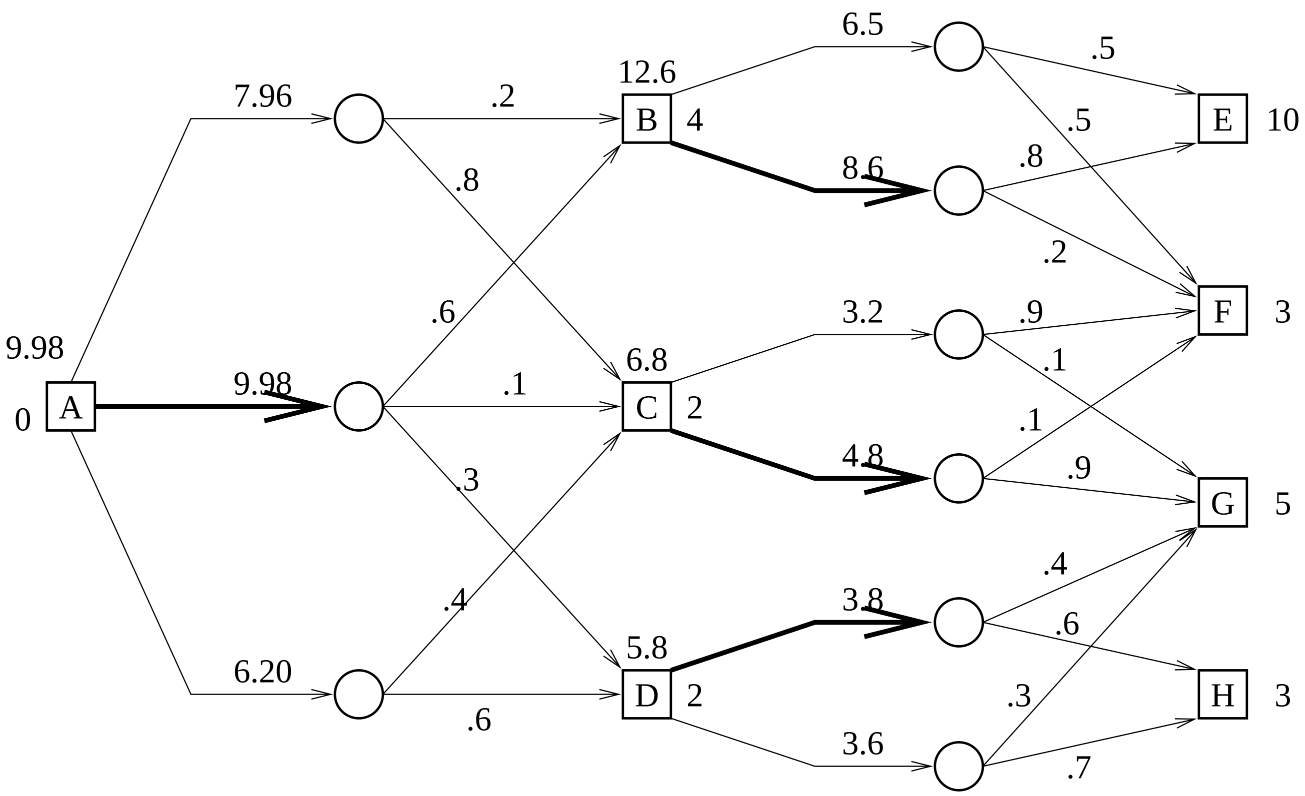
4. Consider the following stochastic reinforcement learning problem. The square nodes are the states (where our policy must choose which action to perform). These are also sometimes called the choice nodes. The circular nodes are random nodes, where a random choice is made (according to the indicated probabilities) to determine the next state. The numbers next to each state specify the reward we get when we enter that state. Compute the optimal value function for this problem. When you perform a backup, you must compute an expected value according to the formula:

$$f(s) = \max_a \sum_{s'} p(s'|s,a)[r(s') + f(s')]$$

where s' is the resulting state after choosing action a. $p(s'|s,a)$ is the probability that if we start in state s and apply operator a we will move to state s'.



We can compute this optimal policy by working backward, taking expected values at each circle node and taking maximums at each square node. The optimal policy and value function are shown here:



6.5
.5
12.6
7.96
.2
B
4
.5
E
10
8.6
.8
.8
.2
.6
3.2
.9
F
3
9.98
6.8
.1
9.98
.1
A
C
2
0
.1
4.8
.9
.3
G
5
.4
3.8
.4
.6
5.8
D
2
H
3
6.20
.3
.6
3.6
.7