Data Mining: Assignment 3

Due date: March 19 (Tuesday)

Problem 1 (5 points for CIS 4930, 3 points for CIS 6930)
Suppose we are trying to learn the concept of "scientist" based on the following examples:

	IQ	good hacker?	has publications?	hobby
positive	160	yes	no	sci-fi
positive	100	yes	no	music
positive	100	yes	yes	tennis
positive	160	no	yes	tennis
positive	130	yes	yes	tennis
positive	100	yes	yes	music
positive	130	no	yes	music
positive	160	no	no	music
positive	160	yes	yes	sci-fi
negative	130	yes	no	sci-fi
negative	130	no	no	sci-fi
negative	100	no	yes	tennis
negative	130	yes	no	music
negative	100	no	no	music

Use these data to construct a decision tree; you should compute the information gains to decide which attributes are more important. For each node of the tree, indicate the corresponding information gain.

Problem 2 (5 points for CIS 4930, 3 points for CIS 6930)
Implement a program for building decision trees, assuming that all instances belong to two classes, "positive" and "negative." It should read a file with training examples and test instances, use the training examples to build a tree, and classify the test instances. The only required output is the classification of the instances; it does *not* have to include the tree itself. The input format is as follows:

```
<class> <attribute> <attribute> ... <attribute>
   ...
<class> <attribute> <attribute> ... <attribute>

<attribute> <attribute> ... <attribute>
   ...
<attribute> <attribute> ... <attribute>
```

The training examples are above the blank line, and the test instances are below. Each `<class>` is either "`positive`" or "`negative`," and each `<attribute>` is a string of lower-case letters. The length of a string is at most twenty letters; successive attributes are separated by one or more spaces. For instance, the following file includes three training examples and two test instances:

```
positive  smart    hacker    nopapers  scifi
positive  average  hacker    papers    music
negative  average  nohacker  nopapers  music

average  hacker    papers    music
smart    nohacker  nopapers  scifi
```

Problem 3 (2 bonus points for CIS 4930, 4 regular points for CIS 6930)
If you are taking CIS 4930, this problem is optional, and it does not affect your grade for the assignment. If you solve it, you get 2 bonus points toward your final grade for the course.

Extend your decision-tree program to allow multiple classes, numeric attributes, and unknown attribute values. It should read the file with training examples and test instances, and classify the instances. You may submit one program for Problems 2 and 3, or two separate programs.

The input format is the same as in Problem 2, but we impose fewer restrictions on `<class>` and `<attribute>`. Each `<class>` is a string of lower-case letters that specifies a class name, and each `<attribute>` is either a string or a natural number. The length of a string is at most twenty letters, and the length of a number is at most four digits. If an attribute value is unknown, we specify it by an asterisk ($*$); note that both training examples and test instances may include unknown values. For instance, the following file includes four training examples and two test instances:

```
scientist   160  hacker    papers    scifi
scientist   100  hacker    *         music
techwriter  160  nohacker  papers    scifi
artist      *    nohacker  nopapers  music

100  hacker    *         music
130  nohacker  nopapers  scifi
```