UNIVERSITY OF NOTRE DAME
Association for Computing Machinery
Programming Contest

Problem Sets

26 March 1994

There are 7 problems, given in order of increasing difficulty. Please notify a judge immediately if you do not have all seven problems. There are 19 labeled pages in this handout.

If you have a problem of any kind, please notify a judge immediately.

Judges will NOT give out helpful hints on any of the programs. We will clarify any questions that you have about the problem, but no help will be given. Please see the handout entitled "Official Rules" for a further explanation.

For programming purposes, you may assume that all data will be of the format specified in the problem specification. That is, no invalid data will be used to judge your program.

NOTE: *No* output should have a blank final line!

1 Heads / Tails Probability

The probability of n heads in a row tossing a fair coin is 2^{-n}. Calculate the probability for $n = 100, 150, \ldots, 300$.

1.1 Input

None.

1.2 Output

Print a table of n and 2^{-n} in the following for the given values of n, using the following format:

```
2^-100 = x.xxxe-y
2^-150 = x.xxxe-y
2^-200 = x.xxxe-y
2^-250 = x.xxxe-y
2^-300 = x.xxxe-y
```

where each `x` is a decimal digit and each `y` is a decimal integer with no leading zeros or spaces.

2 Wild Thing

Command interfaces to modern operating systems often include a number of *metacharacters* which provide powerful features that allow a user to accomplish a lot of work without resorting to traditional programming. One of the most common metacharacters is a *wildcard* character.

Wildcard characters are used within filenames to permit the user to generically specify a group of files that a command should act upon. A wildcard character itself matches zero or more characters in the same position as the wildcard.

Suppose a user wanted to print all files ending in `.c`, delete all files beginning with `old_`, and copy *all* files (no matther what they are named) to a floppy disk. With wildcard metacharacters, all of this can be done using only three commands instead of using a separate command for each file involved. The metacharacter sequences for these commands would be as follows:

```
*.c
old_*
*
```

Write a program that will accept an input pattern which may contain wildcard characters and decide whether or not each of the filenames that follow it matches the pattern.

2.1 Input

The input file will contain a series of datasets. Each dataset will consist of a pattern line followed by a list of filenames to compare against the pattern. For the purposes of this program, a filename may consist of any group of one to twenty characters, not including asterisks. None of these characters should receive special treatment (periods, for example, should behave as any other characters matched).

The pattern line may include zero or more wildcard characters. A wildcard character will be represented by an asterisk (*).

The list of filenames will appear one per line following the pattern line. There will be an unknown number of filenames in the list.

All lines of input will be one to twenty characters in length, including the pattern line. The input file will contain an unknown number of datasets, each separated by a blank line.

2.2 Output

The output for this program will be a short report showing the pattern and the filenames from the subsequent list that match the pattern. Filenames that do not match the pattern should not be printed out. The results from each input dataset should be separated by two blank lines. See the example for the exact format.

2.3 Example

Suppose the input file