Program 1

Due Date: Thursday, October 18, 2005   **Points**: 100

One common assumption users of systems make is that they know which command is being executed when they tyoe the name of the command. This assignment asks you to create a tool to check this.

Your program is to be called "whereis". It should print the full path name of all instances of *command* in the search path.

Input

Your program is to take 1 or more command-line arguments, each naming a command. It takes no other input.

Output

Your program is to output the command followed by a colon and a blank-separated list of program names.

Example

Here is an example run for one command (input and output). The "%" is a shell prompt:

```
% where vi
vi: /usr/bin/vi
%
```

Here is an example run for two commands, the second having two versions:

```
% where sh csh
sh: /bin/sh
csh: ./csh /bin/csh
%
```

The second command has two versions, one in the current working directory and the other in the */bin* directory. If the user doesn't know about the first program, it could be a trap!

Suggestions

The search path is a list of directories separated by colons. If any entry is empty, (for example, "::"), the corresponding directory is the current working directory. For example, the search paths "/bin::/usr/bin" and "/bin:.:/usr/bin" are treated the same. When you type a command *without a "/" in it*, the shell looks in each directory in the list, and executes the first program with the same name as the command you typed.

The search path is stored in the environment variable **PATH**. To get it, use the library function *getenv*(3).

Extra Credit

1. (*Easy; 15 points*) If the named command is "–", then read a list of commands from the standard input. The list consists of one command per line. For each command, print the same information that would be printed if the command were typed as the argument to "whereis".
2. (*Difficult; 50 points*) If the command is built into the user's login shell (as named in the **SHELL** environment variable), or an alias defined in the shell's start-up file(s), print that information also.