Need data from the web - you want Perl

1. Write a Perl program which reads from STDIN a username then a password (not both simultaneously).

   It should then check the password matches one stored for user *username* in the file `users/` *username* `.password` .

   It should print a message indicating the user is unknown if this file does not exist otherwise print an appropiate message indicating whether the password matches.

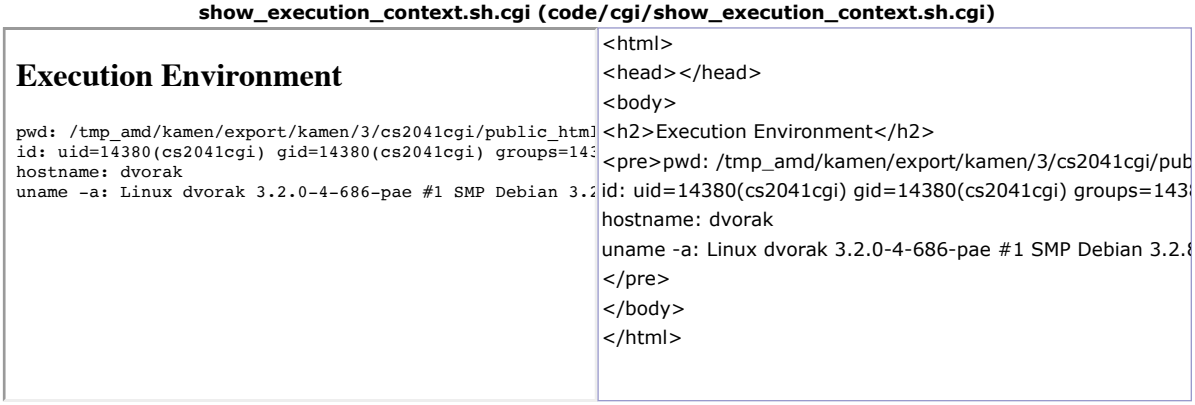
   For example:

```
$ mkdir users
$ echo 'correct horse battery staple' >users/andrewt.password
$ login.pl
username: jas
password: beer
Unkown username!
$ login.pl
username: andrewt
password: 42
Incorrect password!
$ login.pl
username: andrewt
password: correct horse battery staple
You are authenticated.
```

2. How are CGI scripts run?

3. Write a shell CGI script which prints details of the context in which it is run. Here is an example implementation:

   show_execution_context.sh.cgi (code/cgi/show_execution_context.sh.cgi)



Execution Environment

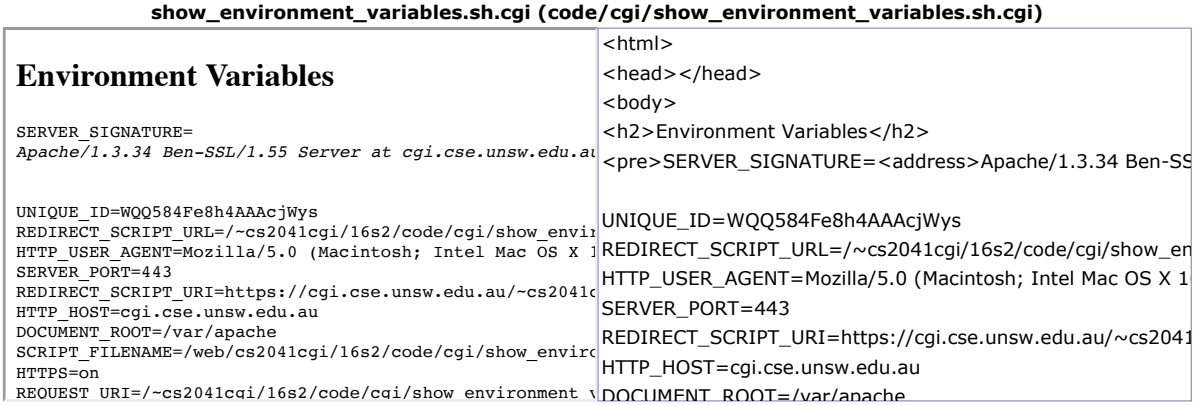
```
pwd: /tmp_amd/kamen/export/kamen/3/cs2041cgi/public_html
id: uid=14380(cs2041cgi) gid=14380(cs2041cgi) groups=143
hostname: dvorak
uname -a: Linux dvorak 3.2.0-4-686-pae #1 SMP Debian 3.2
```

```
<html>
<head></head>
<body>
<h2>Execution Environment</h2>
<pre>pwd: /tmp_amd/kamen/export/kamen/3/cs2041cgi/pub
id: uid=14380(cs2041cgi) gid=14380(cs2041cgi) groups=143
hostname: dvorak
uname -a: Linux dvorak 3.2.0-4-686-pae #1 SMP Debian 3.2.8
</pre>
</body>
</html>
```

   Discuss how security concerns might have affect CSE's choices for configuring how student CGI scripts are run on its web server.

4. Web servers pass some useful information to CGI scripts as environment variables.
   Write a shell CGI script which prints details of the environment variables it has been passed. Here is an example implementation:

   show_environment_variables.sh.cgi (code/cgi/show_environment_variables.sh.cgi)



Environment Variables

```
SERVER_SIGNATURE=
Apache/1.3.34 Ben-SSL/1.55 Server at cgi.cse.unsw.edu.au

UNIQUE_ID=WQQ584Fe8h4AAAcjWys
REDIRECT_SCRIPT_URL=/~cs2041cgi/16s2/code/cgi/show_envi
HTTP_USER_AGENT=Mozilla/5.0 (Macintosh; Intel Mac OS X 1
SERVER_PORT=443
REDIRECT_SCRIPT_URI=https://cgi.cse.unsw.edu.au/~cs2041c
HTTP_HOST=cgi.cse.unsw.edu.au
DOCUMENT_ROOT=/var/apache
SCRIPT_FILENAME=/web/cs2041cgi/16s2/code/cgi/show_envirc
HTTPS=on
REQUEST_URI=/~cs2041cgi/16s2/code/cgi/show_environment_v
```

```
<html>
<head></head>
<body>
<h2>Environment Variables</h2>
<pre>SERVER_SIGNATURE=<address>Apache/1.3.34 Ben-SS

UNIQUE_ID=WQQ584Fe8h4AAAcjWys
REDIRECT_SCRIPT_URL=/~cs2041cgi/16s2/code/cgi/show_en
HTTP_USER_AGENT=Mozilla/5.0 (Macintosh; Intel Mac OS X 1
SERVER_PORT=443
REDIRECT_SCRIPT_URI=https://cgi.cse.unsw.edu.au/~cs2041
HTTP_HOST=cgi.cse.unsw.edu.au
DOCUMENT_ROOT=/var/apache
```

   Discuss what some of these environment variables might indicate.

5. A HTTP request can pass parameters to a web server. The web server passes these on to the CGI script typically in environment variable(s) for GET request and on STDIN for POST requests.

Write a shell CGI script which prints whether it was invoked with a GET request or a POST request and prints details of the parameters it has been passed. Here is an example implementation:

show_input_parameters.sh.cgi?comp2041=best-course-ever (code/cgi/show_input_parameters.sh.cgi?comp2041=best-course-ever)

GET Request - Input Parameters
 	
`comp2041=best-course-ever`
	<html>
	<head></head>
	<body>
	
	 GET Request - Input Parameters
	<pre>comp2041=best-course-ever
	</pre>
	</body>
	</html>

6. A HTTP request can pass pararmeters to a web server. The web server passes these on to the CGI script typically in environment variable(s) for GET request and on STDIN for POST requests
   Write a shell CGI script which (given no parameters) prints a form allowing 2 numbers to be entered. The form should run the same CGI script, and the CGI script should print the sum of the numbers and a form allowing two more numbers to be entered and the script to be run again. Here is an example implementation:

sum_two_numbers.sh.cgi (code/cgi/sum_two_numbers.sh.cgi)

+ = ? calculate	<html> <head> <title>Sum Two Numbers</title> </head> <body> <form method="GET" action=""> <input type="textfield" name="x" value=""> + <input type="textfield" name="y" value=""> = ? <input type="submit" value="calculate">