NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING
EXAMINATION FOR
Semester 2, 2013/2014

CP3101B - Web Programming and Applications

30 April 2014 Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. The examination paper contains **SIX (6) questions** and comprises **TWENTY (20) pages.**
2. Weightage of questions is given in square brackets. The maximum attainable score is 50.
3. This is a **CLOSED** book examination, but you are allowed to bring **TWO** double-sided A4 sheets of notes for this exam.
4. Write all your answers in the space provided in this booklet.
5. Please pull the Appendix from the rest of the exam, for easier reference.
6. **Please write your matriculation number below.**

MATRICULATION NUMBER: _____

(this portion is for the examiner's use only)

Question	Marks	Remark
Q1		
Q2		
Q3		
Q4		
Q5		
Q6		
Total		

Question 1: HTTP etc. [10 marks]

This question involves `login.php` and `add.php` (page 13) in the appendix to this exam. Both of these scripts are at

`http://cp3101b-1.comp.nus.edu.sg/~arnold/add/`.

A.

Write down a minimal HTTP1.1 request to `login.php` (page 13) and the resulting response which would result in the user being logged into the application. Assume that this is the first HTTP request to this website. Make up any values you need to illustrate the requests and responses.

B.

Now, assuming the user is authenticated as a result of Part A, write down a minimal HTTP1.1 request to `add.php` (page 13) which would have the script add the numbers 2,3,5,7,11. Make sure to include the HTTP response. Make up any values you need to illustrate the requests and responses.

Question 2: HTML5/CSS [10 marks]

Write `style.css` below, to appropriately style `http://cp3101b-1.comp.nus.edu.sg/~arnold/add/index.html` (see the appendix, page 14). Your `style.css` and `index.html` should generate the screenshots displayed in the appendix (pages 15-17).

```
/* style.css is below ------------------------------------------------
   HINT: The only pixel measurements in the css are 0px, 1px, 10px and 50px
   HINT: The colors are black, white, red, grey, lightgrey, green, lightgreen
   HINT: The background color for the result is red when there are errors,
         and green when there are no errors. Capture this in the css.
   HINT: The body has a dashed border. */
```

(continued...)

Question 3: AJAX/Javascript/JQuery [10 marks]

Write `http://cp3101b-1.comp.nus.edu.sg/~arnold/add/application.js` below.
`application.js`, together with `index.html`, `login.php` and `add.php` (pages 13-17) allows a user to login and then repeatedly submit numbers for processing. The final application should behave as outlined in the appendix screenshots (pages 15-17). If it helps, you can modify index.html, but only to modify or add attributes to existing elements.

(continued)

Question 4: PHP, SQL [5 marks]

A web application consists of a single PHP script which uses a Postgresql database as outlined in the **Web Security** appendix at the end of this exam (pages 18-19). In the space below, describe what this application does. Provide a few screenshots to show how it responds to user input.

(continued ...)

Question 5: MVC [5 marks]

For the code below, circle lines or blocks of code and write M,V or C beside them to identify them as primarily concerning the Model, View or Controller. Account for as much code as possible.
Note: The code below is 100% the same as in the **PHP, SQL** question, copied from page 19 for convenience.
Note: The code is definitely not architected as an MVC application, so M,V and C are intermixed.

```
<?php
    require 'config.inc';
    session_save_path("sess");
    session_start();
    if(!isset($_SESSION['loggedin'])){
        $_SESSION['loggedin']='false';
    }
    $dbconn = pg_connect("$connect_string");
    if(!$dbconn){
        echo("Can't connect to the database");
        exit;
    }
    if(isset($_REQUEST['username']) && isset($_REQUEST['password'])) {
        $_SESSION['loggedin']='false';
        $query = "SELECT * FROM appuser WHERE username=$1 AND password=$2;"
        $result = pg_prepare($dbconn, "", $query);
        $result = pg_execute($dbconn, "", array($_REQUEST['username'],$_REQUEST['password']));
        if(pg_num_rows($result)==1){ # found a row, so we can login
            $_SESSION['loggedin']='true';
            $_SESSION['username']=$_REQUEST['username'];
        }
    }
    if($_SESSION['loggedin']=='true' && isset($_REQUEST['num'])){
        $query="INSERT INTO numbers(username, num) VALUES ('$_SESSION[username]', $_REQUEST[num]);");
        $result=pg_query($dbconn, $query);
    }
?>
<?php if($_SESSION['loggedin']=='true'){ ?>
        <form>
            num: <input type="text" name='num' /> <input type="submit" value="add to list" />
        </form>
        <table>
            <tr><th>user</th><th>number</th></tr>
<?php
            $result=pg_query($dbconn, "SELECT * FROM numbers;");
            while ($row = pg_fetch_array($result)) {
                echo("<tr><td>$row[username]</td><td>$row[num]</td></tr>");
            }
?>
        </table>
<?php } else { ?>
        <form>
            User: <input type="text" name='username' /> <br/>
            Password: <input type="password" name="password" />
            <input type="submit" value="login" />
        </form>
<?php } ?>
```

Question 6: SECURITY [10 marks]

A cp3101b-1 web application consists of a single PHP script which uses a Postgresql database as outlined in the **Web Security** appendix (pages 18-19) at the end of this exam (same code as in the last two questions). Detail all of the security issues you find with this application. Give explicit examples and exploits where possible. For example, if it is vulnerable to SQL Injection, give an explicit example. Similarly for XSS and file/directory permission issues. For each vulnerability, explain who can exploit it and how. **NOTE:** Please review all files in the **Web Security** appendix of this exam for this question.

FILE PERMISSION ISSUES (complete the table below)...

```
The permissions on files are wrong, they should be ..., and here is why ...
drwx--x--x     webSecurity
    issue: www-data does need to traverse this directory to read .htaccess,
           it does not need to ls this directory, so 'r'
           privilege is not needed.

-rw-------     webSecurity/config.inc
    issue:

               webSecurity/index.php
    issue:

               webSecurity/schema.sql
    issue:

               webSecurity/sess
    issue:
```

(continued ...)

XSS, SQL INJECTION and OTHER ISSUES ...

Appendix (AJAX)

login.php

```
<?php
        session_save_path("sess");
        session_start();
        header('Content-Type: application/json');

        $reply=array('status'=>'invalid login');
        $_SESSION['loggedin']='false';

        if(empty($_REQUEST['user'])||empty($_REQUEST['password'])){ goto leave; }

        $credentials = json_decode($_REQUEST['params'], true);

        if($_REQUEST['user']=='arnold' && $_REQUEST['password']=='spiderman'){
                $reply['status']='ok';
                $_SESSION['loggedin']='true';
        }
        leave: print json_encode($reply);
?>
```

add.php

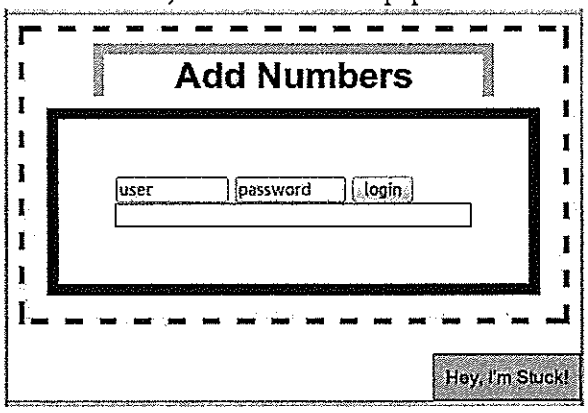
```
<?php
        session_save_path("sess");
        session_start();
        header('Content-Type: application/json');

        $reply=array('status'=>'ok');

        if(!isset($_SESSION['loggedin']) || $_SESSION['loggedin']!='true'){
                $reply['status']='not authorized';
                goto leave;
        }
        if(empty($_REQUEST['params'])){ goto leave; }

        $numbers = json_decode($_REQUEST['params'], true);
        $sum=0;
        for($i=0;$i<count($numbers);$i++){
                $num=$numbers[$i];
                if(!is_numeric($num)){
                        $reply['status']='invalid input';
                        goto leave;
                }
                $sum=$sum+$num;
        }
        $reply['sum']=$sum;

        leave: print json_encode($reply);
?>
```

index.html

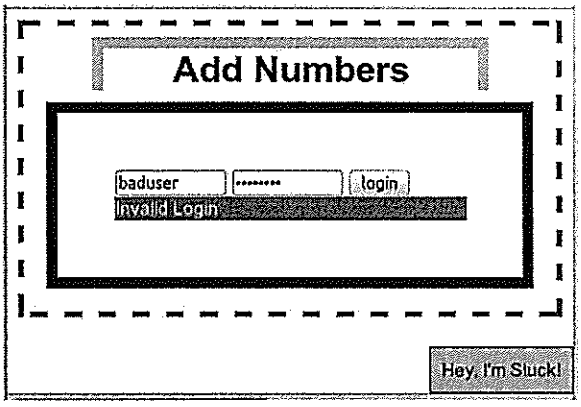
```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <link rel="stylesheet" type="text/css" href="style.css">
        <script src="http://code.jquery.com/jquery-latest.min.js"></script>
        <script src="application.js"> </script>
    </head>
    <body>
        <header> <h1> Add Numbers </h1> </header>
        <section>
            <div id="login_gui">
                <form>
                    <input type="text" id="user" size="5" placeholder="user" required />
                    <input type="password" id="password" size="5" placeholder="password" required />
                    <input type="button" value="login" />
                </form>
            </div>
            <div id="add_gui" style="display:none;">
                Give me a list of numbers:
                <form id="numbers_form">
                    <input type="text" size="1" data-num="0" />
                    <input type="text" size="1" data-num="1" />
                    <input type="text" size="1" data-num="2" />
                    <input type="text" size="1" data-num="3" />
                    <input type="text" size="1" data-num="4" />
                    <input type="button" value="add" />
                </form>
            </div>
            <div id="result">   </div>
        </section>
        <footer>
            Hey, I'm Stuck!
            <!-- lightgreen, stuck at the bottom right of the browser window -->
        </footer>
    </body>
</html>
```

Screenshots

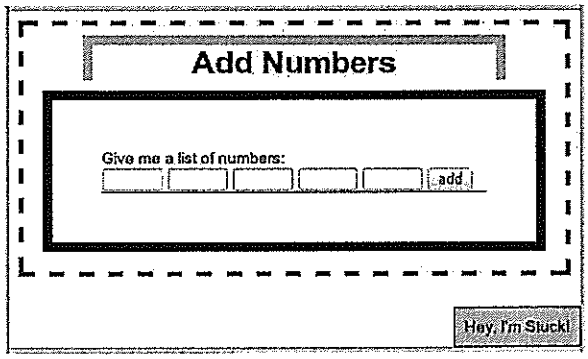
1) First visit to index.php.



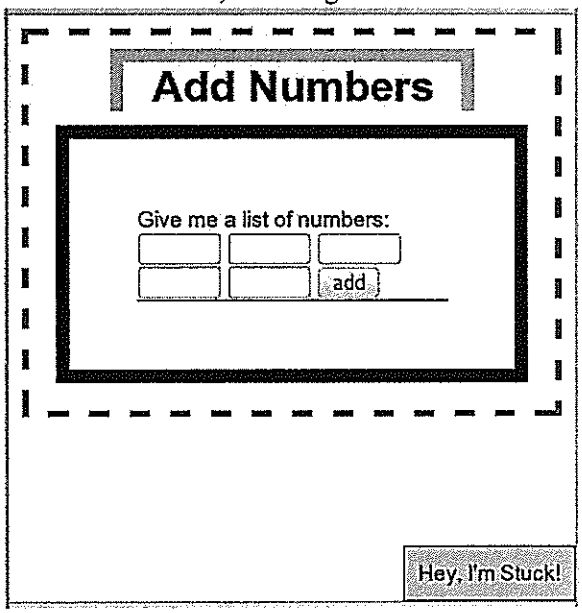
2) After submission of bad username/password.



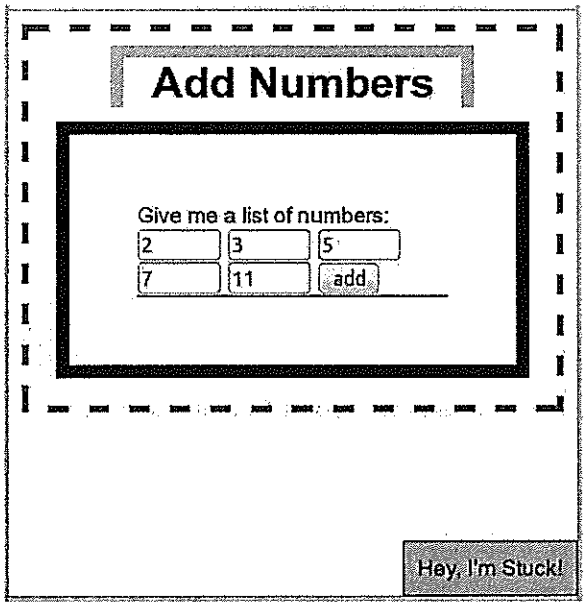
3) Login Success. Resize window.



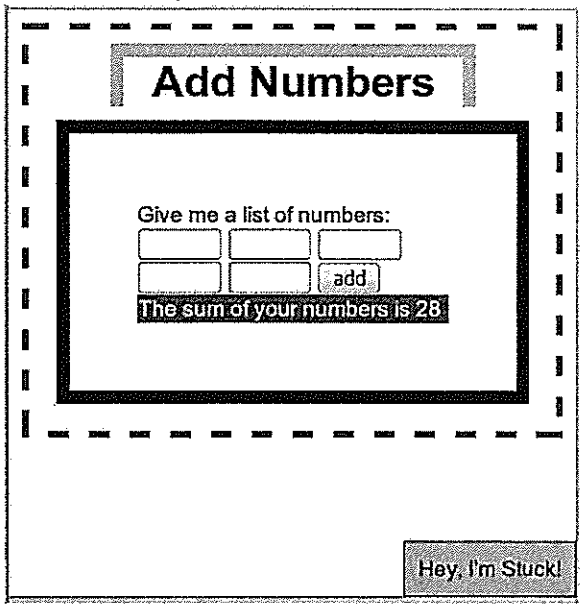
4) Resize again.



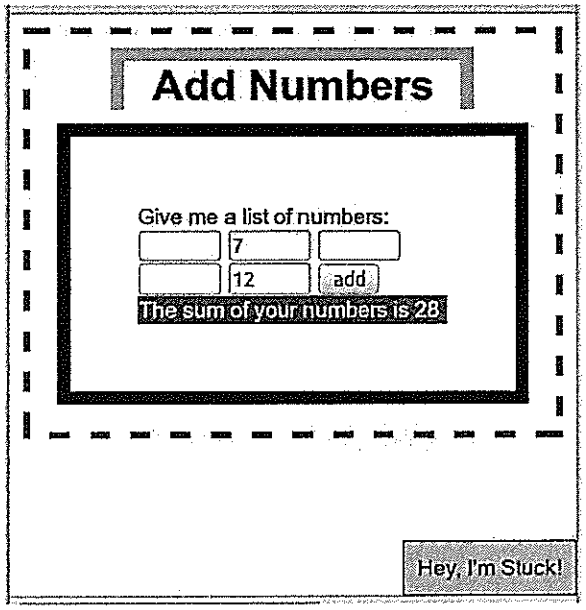
5) Put in some numbers, about to submit.



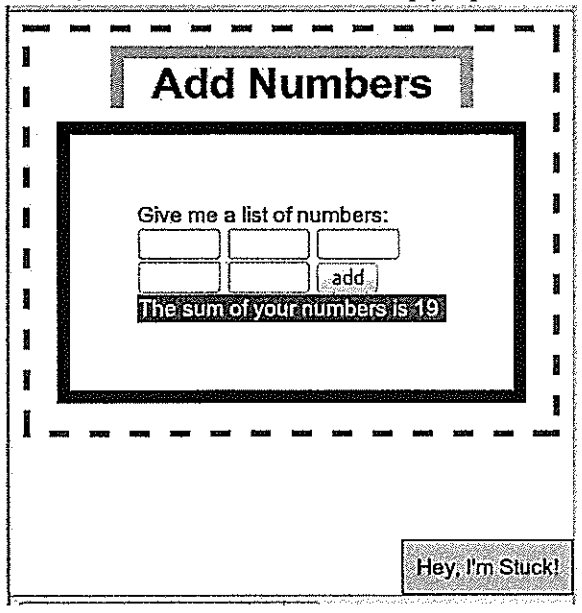
6) Result of submission.



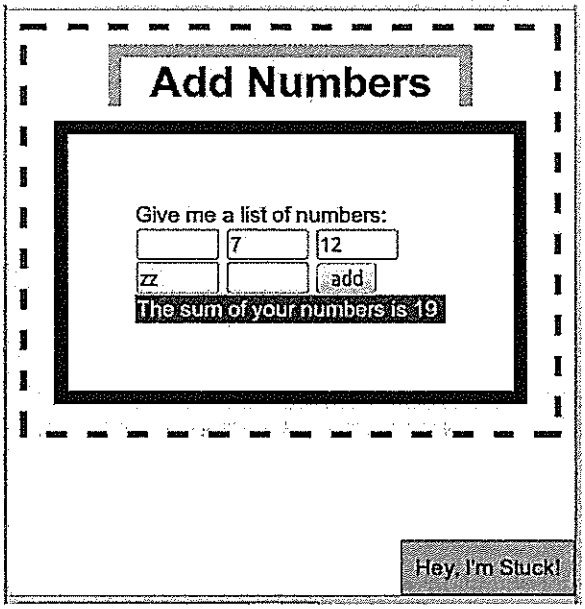
7) About to submit, some fields left blank.



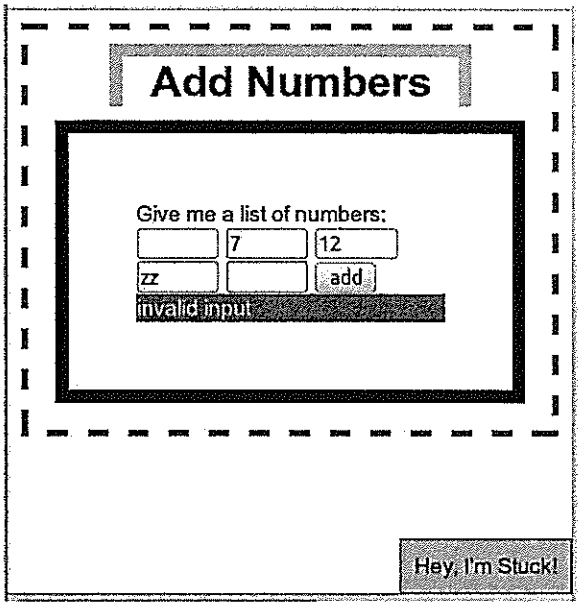
8) Result of submission with empty inputs.



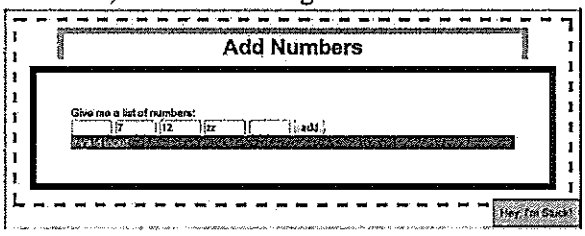
9) About to submit, with invalid field (ZZ).



10) Result of submission with invalid field.



11) Result of resizing browser window.



Appendix (Web Security)

file permissions

```
drwxr-xr-x webSecurity
-rw-r--r-- webSecurity/config.inc
-rw-r--r-- webSecurity/index.php
-rw-r--r-- webSecurity/schema.sql
drwxr-xr-x webSecurity/sess
```

webSecurity/schema.sql

```
DROP TABLE appuser CASCADE;
DROP TABLE numbers CASCADE;
CREATE TABLE appuser (
    username VARCHAR(20) PRIMARY KEY,
    password VARCHAR(20) NOT NULL
);
INSERT INTO appuser (username, password) VALUES('sid', 'asjdijijs');
INSERT INTO appuser (username, password) VALUES('sally', 'aajsutunnd');
INSERT INTO appuser (username, password) VALUES('jill', '772jjs79');
INSERT INTO appuser (username, password) VALUES('joe', 'ww87q332');
CREATE TABLE numbers (
    username VARCHAR(20) REFERENCES appuser(username),
    num INTEGER NOT NULL
);
```

webSecurity/config.inc

```
<?php
$db_user='tester';
$db_name='tester';
$db_password='password';
$connect_string="host=127.0.0.1 port=5432 dbname=$db_name user=$db_user password=$db_password";
?>
```

webSecurity/index.php

```
<?php
    require 'config.inc';
    session_save_path("sess");
    session_start();
    if(!isset($_SESSION['loggedin'])){
        $_SESSION['loggedin']='false';
    }
    $dbconn = pg_connect("$connect_string");
    if(!$dbconn){
        echo("Can't connect to the database");
        exit;
    }
    if(isset($_REQUEST['username']) && isset($_REQUEST['password'])) {
        $_SESSION['loggedin']='false';
        $query = "SELECT * FROM appuser WHERE username=$1 AND password=$2;"
        $result = pg_prepare($dbconn, "", $query);
        $result = pg_execute($dbconn, "", array($_REQUEST['username'],$_REQUEST['password']));
        if(pg_num_rows($result)==1){ # found a row, so we can login
            $_SESSION['loggedin']='true';
            $_SESSION['username']=$_REQUEST['username'];
        }
    }
    if($_SESSION['loggedin']=='true' && isset($_REQUEST['num'])){
        $query="INSERT INTO numbers(username, num) VALUES ('$_SESSION[username]', $_REQUEST[num]);");
        $result=pg_query($dbconn, $query);
    }
?>
<?php if($_SESSION['loggedin']=='true'){ ?>
        <form>
            num: <input type="text" name='num' /> <input type="submit" value="add to list" />
        </form>
        <table>
            <tr><th>user</th><th>number</th></tr>
<?php
            $result=pg_query($dbconn, "SELECT * FROM numbers;");
            while ($row = pg_fetch_array($result)) {
                echo("<tr><td>$row[username]</td><td>$row[num]</td></tr>");
            }
?>
        </table>
<?php } else { ?>
        <form>
            User: <input type="text" name='username' /> <br/>
            Password: <input type="password" name="password" />
            <input type="submit" value="login" />
        </form>
<?php } ?>
```

— E N D O F P A P E R —

Scratch Paper

HAVE A GREAT SUMMER!