**COMP 3540 Advanced Web Design and Programming**

Final Exam – Fall 2016

**Two and a half hours (2.5 hours)**

Student Name: Student Number:

Note 1. Do not include any unnecessary explanations or guesses in your answers. They will get **negative** marks.

Note 2. Answer clearly, completely, and correctly. In general, partial marks will **not** be given.

1. (8 marks) Fill in the blanks in each of the following.
2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ are patterns used to match character combinations in strings.
3. The PHP \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ function searches a string for pattern, returning true if the pattern exists, and false otherwise.
4. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ attribute specifies that an input field must be filled out before submitting the form.
5. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is a web development design pattern that prevents some duplicate form submissions.
6. With \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, not cookies, web applications can store data locally within the user’s browser.
7. When the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ attribute is present, the user is allowed to enter more than one value in the input element.
8. (2 marks) HTML5 introduced the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_() that allows you to add history entries in your web browser.
9. (4 marks) Complete the following code that can submit multiple files.

<form action='file\_upload.php' method='\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_'

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_='\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_'>

<input type='\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_' name='\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_' \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_>

<button type='submit'>Upload files</button>

</form>

1. (4 marks) Complete the following PHP code that receives the files submitted from the code in 2).

for ($i = 0; $i < count(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_[\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_][\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]); $i++) {

$filename = basename(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_[\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_][\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_][\_\_\_\_\_\_\_\_]);

move\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_[\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_][\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_][\_\_\_\_\_\_\_\_],

"uploads/" . $filename);

}

1. (4 marks) Complete the following code that uses drag and drop. (Hint: Here are related events – dragstart, dragover, drop.)

<img id="img1" src="logo.gif" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_="true"

on\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_="**drag**(event)" width="80" height="60">

<div id="div1" on\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_="**allow\_drop**(event)" on\_\_\_\_\_\_\_\_\_="**drop**(event)" ></div>

<script>  
 function **allow\_drop**(ev) {

ev.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(); }  
 function **drag**(ev) {

ev.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.setData("id", ev.target.id); }  
 function **drop**(ev) {  
     ev.preventDefault();  
     var id = ev.dataTransfer.getData("id");  
     ev.target.appendChild(document.getElementById(id));  
 }  
</script>

1. (4 marks) Complete the following code that stores data locally within the user’s browser and retrieves the data later. You should not use cookies. Note that the data should not be erased when the user’s browser is closed, so that the data can be retrieved later when the code runs again.

var person = {name: "Smith", age: 20};

// Convert the above object to a string.

person = JSON.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(person);

// Save person into the storage with the key value “p1”.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.setItem(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, person);

// Some time later, retrieve the string from the storage with the same key,

// convert the string to an object, and print the object.

var person = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

person = JSON.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(person);

alert(person.name + "; " + person.age);

1. (2 marks) Complete the following code that prevents the resubmission of username and password in controller.php for TRUQA.

'SignIn':

if (isValid($username, $password)) // If username and password are valid

// Redirect to ‘view\_main.php’.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_('\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_: ' . 'view\_main.php');

Else

// Redisplay the start page

include('view\_start.php');

1. (4 marks) Complete the following code that changes the background color of the content loaded into an iframe with id='id\_iframe'.

$(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_).\_\_\_\_\_\_\_\_\_\_('load', \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_() {

this.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.document.body.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.backgroundColor = 'Cyan';

});

1. (6 marks) Write the jQuery code that sends an object using POST to ‘controller.php’. The object to be sent includes three properties with their values as follows.

* command: ‘SignIn’
* name: the value stored in the variable username
* password: the value stored in the variable password

The code receives data back from ‘controller.php’, and simply displays the data into the HTML element with id=‘result-pane’. (You should not use a form. You should not include anything unnecessary.)

1. (4 marks) Give a regular expression for passwords that satisfy the followings.
   * Minimum 6 characters should be included.
   * At least one digit should be included.
   * Password should start with a lower case letter.
2. (2 marks) Give an example of SSE message of ‘COMP 3540 is an amazing course!’ that is sent from server-side script as data (not retry, event, id). You do NOT have to write the PHP code. Just a string with proper format is enough.

For the questions 11 and 12: There are three tables ‘Users’, ‘Questions’, and ‘Answers’ in a MySQL database.

* The ‘Users’ table has three columns – Id, Username, Password
* The ‘Questions’ table has three columns – QuestionId, UserId, Question
* The ‘Answers’ table has four columns – AnswerId, QuestionId, Answer, UserId

1. (4 marks) Write the PHP code that

* Gets all the questions posted by the user whose name is submitted from the client,
* Saves the questions into an array,
* Converts the array into a JSON string, and
* Sends the string back to the client.

// Use $conn to access the db.

$username = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_['username'];

$sql = ;

$result = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_($conn, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_);

$data = array();

$i = 0;

while($row = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_($result)) {

$data[$i] = ;

$i++;

}

$str = json\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_($data);

echo $str;

1. (4 marks) Write the PHP code that

* Searches questions that include a search term sent from the client,
* Gets all the answers for the questions,
* Saves the answers into an array, with the corresponding question id, (Hint: array\_merge(array1, array2) can be used to merge two associative arrays. This function returns the new array.)
* Converts the array into a JSON string, and
* Sends the string back to the client.

// Use $conn to access the db.

$search\_term =

$sql =

$result =

$data =

$i = 0;

while($row = ) { // for each question

$sql =

$result\_answers = mysqli\_sql(

while($r = ) { // for each answer

$data[ ] =

$i\_\_\_\_\_\_\_;

}

}

$str =

echo $str;

1. (4 marks) Complete the following code that writes HTML table code for the data sent from the code in 12).

var obj = JSON.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(str); // linear array of associative arrays

var str = '<table>';

for (var i=0; i < obj.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; i++) {

str += '<tr>';

for (var p \_\_\_\_\_\_\_\_\_\_ obj[i]) {

str += '<td>' + \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ + '</td>';

}

str += '</tr>';

}

str += \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;