CS380 Information Retrieval and Web Search

Lab 5

PHP and Sessions

Monday, March 2

You may work in small groups for this lab.

Writing PHP that tracks (fills a simple shopping cart):

In this lab you will write a PHP page (program) that records the session history of inputs from a form (think of a shopping cart). You will be using session tracking in PHP as discussed in class on Wednesday and your form from the previous lab.

Step 1: Create a PHP page to receive your input from your form. Some of the following PHP code, from Wednesday's lecture, will be useful. Include somewhere in your PHP page a link back for your form.

```
<html><body>
HEADERS<br>
<?php
foreach (getallheaders() as $name => $value) {
    echo "$name: $value<br>";
}
?>
POST ARGS
<?php
foreach ($_POST as $name => $value) {
    echo "POST $name: $value<br>";
}
</body></html>
```

Step 2: Add the data submitted by your form to the $_SESSION variable. **Do not overwrite data already in the session**, add it to the session variable with a new name. It might be useful here to have a count of the number of times the PHP page has been hit in this session. For instance, the following PHP code will do that.

```
$count=1;
if (array_key_exists("count", $_SESSION)) {
    $count = $_SESSION["count"]+1;
}
$_SESSION["count"]=$count;
```

Step 3: Show the entire contents of the session to date (use the PHP foreach loop). This should be all of the input from every time you submitted from your form (in this session). Put this output into a html table.

Step 4: Verify that sessions in PHP work. For instance, have two sessions hitting you PHP page at the same time. What other tests do you need to validate PHP sessions?

<u>Turn in:</u>

1. Your PHP code.
2. A description of the tests to verify that PHP sessions work. (This could be hand written)
3. The cookie that PHP set for one of your sessions
4. Answers to the following questions:
    1. What happens to your session tracking when you block cookies in Chrome?
    2. Are sessions shared across tabs?