

**Computer Science 571 Exam #1**  
**Prof. Papa**  
**Thursday, October 11, 2013, 5:30pm – 6:45pm**

**Name:**

**Social Security or Student Id Number:**

1. This is a closed book exam.
2. Please answer all questions.
3. Place all answers on the exam and return the entire exam

**PHP Questions [20 pts]**

**Each question is worth 4 points.**

**Q1: Which of the following is true in PHP?**

**A1:**

- ☐ PHP is strictly typed
- ☐ Function names are case-sensitive
- ☐ Identifiers can contain spaces.
- ☐ Local variables are restricted to a function or class
- ☐ Variable names begin with a dollar sign (\$) followed by underscore ( \_ )
- ☐ ALL OF THE ABOVE
- ☐ NONE OF THE ABOVE

**Q2: What is the meaning of the `$_SERVER[ 'PHP_SELF' ]` variable?**

**A2:**

**Q3: Consider the PHP code snippet:**

```
<?php
$foo = 25;           // Numerical variable
$bar = "Hello";      // String variable
$bar = ($foo * 7) + $bar; //valid or invalid expression?
?>
```

**Is the expression on the third line of the script valid or invalid?**

**A3:**

**Q4: Consider this PHP code snippet:**

```
<?php
$foo = 25;
$bar = "Hello";      // String variable
```

```
echo $foo,$bar;  
echo "5x5=", $foo;  
?>
```

**What is the output produced?**

**A4:**

**Q5: Assume a form contains one textbox named "myName", and the form is submitted using the GET method. How would a PHP script access the form data?**

**A5:**

## **JAVASCRIPT REGULAR EXPRESSIONS QUESTIONS [10 PTS]**

JavaScript 1.5 introduced a new notation for Regular Expressions using the the ?= notation.

**Q1 (8 pts).** Please write a regular expression that restricts a password to at least one number, one lowercase and one uppercase letter and at least 8 characters that are letters, numbers or the underscore.

**A1.**

Note: there is NO partial credit provided for this question.

**Q2 (2 pts).** What additional attributes are introduced in HTML5 for the form input fields to make it easier to demand and match the input?

**A2.**

## **DOM QUESTIONS [20 PTS]**

Below is the Teaching Assistant's solution to homework #4. You should recall that homework #4 was the XML exercise that involved writing an HTML/JavaScript program, which takes the URL of an XML document containing information related to books, parses the XML file, and displays results in HTML. The resulting HTML will display an HTML table showing the data for the books from the XML file and the books image. Some of the code has been removed. Your task is to restore the missing code. Place your answer to the right of the #XXXXXXs. There are 10 questions each worth 2 points.

```

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Homework #4</title>
</head>
<!--
<body>
</body>
-->

<body>
  <script type="text/javascript">
    function loadXML(url) {
      var xmlhttp;
      if (window.XMLHttpRequest) {
        xmlhttp = new XXXXXXXXX1;

      } else {
        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
      }
      xmlhttp.open("GET", url, false);
      xmlhttp.XXXXXXXX2;

      return xmlhttp;
    }

    function query() {
      var input = XXXXXXXXX3;

      var xmlhttp;
      var xmlDoc;
      var bookNodes;
      var count;
      var i;
      var e;
      var w;
      if (!input) {
        alert("Please input XML file name.");
        return;
      }
      xmlhttp = XXXXXXXXX4;

      if (xmlhttp.status == 404) {
        alert("Cannot find file.");
        return;
      } else if (xmlhttp.status != 200) {
        alert("Communication error.");
        return;
      }
      xmlDoc = xmlhttp.responseXML;
      if (!xmlDoc || xmlDoc.documentElement.nodeName ==
"parsererror") {
        alert("Error in XML file.");
        return;
      }
    }
  </script>
</body>

```

```

        } else {
            bookNodes =
xmlDoc.documentElement.getElementsByTagName("book");
            count = 0;
            for (i = 0; i < bookNodes.length; i++) {
                if (bookNodes[i].parentNode ==
xmlDoc.documentElement
                                && bookNodes[i].childNodes.length >
1) {
                                    count++;
                                    break;
                                }
                            }
                    if (!count) {
                        alert("Error in XML file.");
                        return;
                    }
                }
            w = window.open("", "",
"width=800,height=600,scrollbars=yes");
            w.document
                .write("<table border='1' style='text-align:
center;'>

                <tr><th>author</th><th>title</th><th>genre</th><th>price</th>

                <th>publish_date</th><th>description</th><th>image</th></tr>");
                for (i = 0; i < XXXXXXXXX5; i++) {

                    if (bookNodes[i].childNodes.length < 2) {
                        continue;
                    }
                    w.document.write("<tr style='text-align:
left;'><td>");

                    e = XXXXXXXXX6;

                    if (e && e.firstChild &&
e.firstChild.nodeValue.trim()) {
                        w.document.write(e.firstChild.nodeValue);
                    } else {
                        w.document.write("NA");
                    }
                    w.document.write("</td><td>");
                    e = bookNodes[i].getElementsByTagName("title")[0];
                    if (e && e.firstChild &&
e.firstChild.nodeValue.trim()) {
                        w.document.write(e.firstChild.nodeValue);
                    } else {
                        w.document.write("NA");
                    }
                    w.document.write("</td><td>");
                    e = bookNodes[i].getElementsByTagName("genre")[0];
                    if (e && e.firstChild &&
e.firstChild.nodeValue.trim()) {
                        w.document.write(e.firstChild.nodeValue);
                    } else {
                        w.document.write("NA");
                    }

```

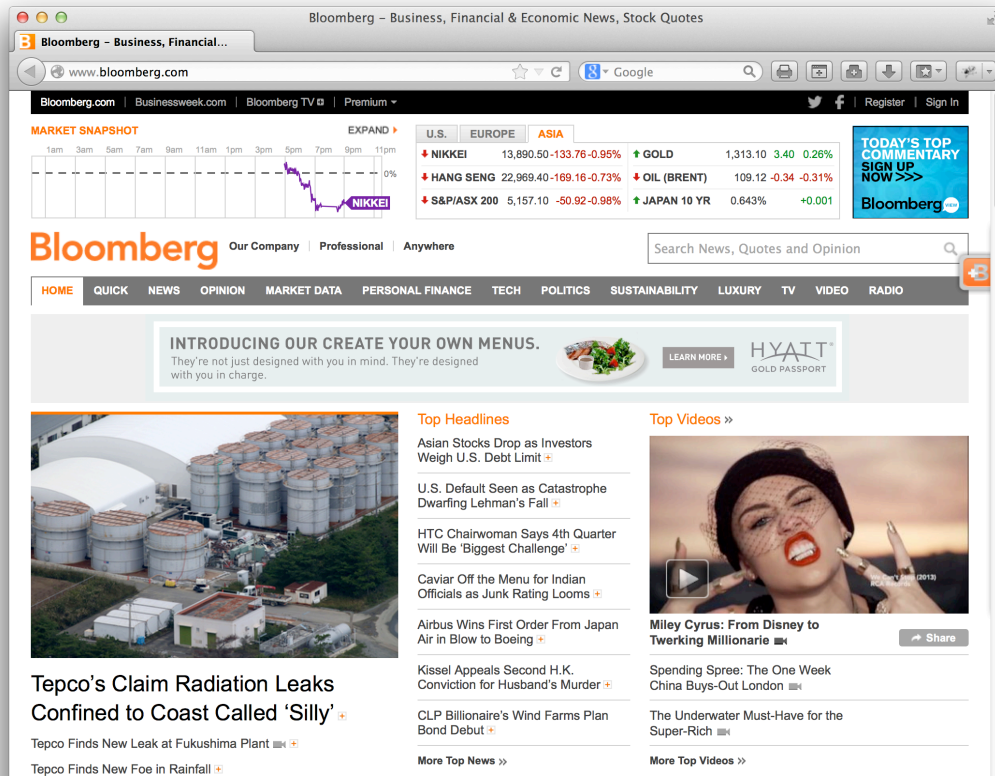
```

        }
        w.document.write("</td><td>");
        e = bookNodes[i].getElementsByTagName("price")[0];
        if (e && e.firstChild &&
e.firstChild.nodeValue.trim()) {
            w.document.write(e.firstChild.nodeValue);
        } else {
            w.document.write("NA");
        }
        w.document.write("</td><td>");
        e =
bookNodes[i].getElementsByTagName("publish_date")[0];
        if (e && e.firstChild &&
e.firstChild.nodeValue.trim()) {
            w.document.write(e.firstChild.nodeValue);
        } else {
            w.document.write("NA");
        }
        w.document.write("</td><td>");
        e =
bookNodes[i].getElementsByTagName("description")[0];
        if (e && e.firstChild &&
e.firstChild.nodeValue.trim()) {
            w.document.write(e.firstChild.nodeValue);
        } else {
            w.document.write("NA");
        }
        w.document.write("</td><td>");
        e = bookNodes[i].getElementsByTagName("image")[0];
        if (e && e.firstChild) {
            w.document
                .write(XXXXXXXX7
                + " alt='Image Not
Available'>")
        } else
            w.document.write("Image Not Available");
        }
        w.document.write("</td></tr>");
    }
    w.document.close();
}
</script>
<center>
    Enter URL for Book XML File <br> <input XXXXXXXXX8 XXXXXXXXX9
    size="50"> <br>
    <button onclick="XXXXXXXX10;">Submit Query</button>
</center>
</body>
</html>

```

## HTTP QUESTIONS [20 PTS]

Below is the web page [www.bloomberg.com](http://www.bloomberg.com). Following that is part of the exchange of messages between the browser and the server, obtained by running Live HTTP Headers. Answer the questions that follow.



(1) <http://bloomberg.com/>

```
GET / HTTP/1.1
Host: bloomberg.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)
Gecko/20100101 Firefox/21.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
```

```
HTTP/1.1 301 Moved Permanently
Date: Mon, 07 Oct 2013 04:26:05 GMT
Server: Apache
Location: http://www.bloomberg.com/
Content-Type: text/html; charset=iso-8859-1
Vary: Accept-Encoding
Content-Encoding: gzip
```

Content-Length: 193  
Keep-Alive: timeout=5, max=500  
Connection: Keep-Alive

-----  
(2) http://www.bloomberg.com/

GET / HTTP/1.1  
Host: www.bloomberg.com  
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)  
Gecko/20100101 Firefox/21.0  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
DNT: 1  
Connection: keep-alive

HTTP/1.1 200 OK  
Status: 200 OK  
Content-Type: text/html; charset=utf-8  
X-Runtime: 9  
Etag: "427fcf8f86c48aee41969a854ac93363"  
Vary: Accept-Encoding  
Content-Encoding: gzip  
Content-Length: 38719  
Cache-Control: private, max-age=51  
Date: Mon, 07 Oct 2013 04:26:06 GMT  
Connection: keep-alive

-----  
(3) http://cdn.gotraffic.net/v/20131004\_173359/stylesheets/main-min-v3\_1.css

GET /v/20131004\_173359/stylesheets/main-min-v3\_1.css HTTP/1.1  
Host: cdn.gotraffic.net  
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)  
Gecko/20100101 Firefox/21.0  
Accept: text/css,\*/\*;q=0.1  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
DNT: 1  
Connection: keep-alive

HTTP/1.1 200 OK  
Server: Sun-Java-System-Web-Server/7.0  
Vary: Accept-Encoding  
Expires: Sat, 04 Oct 2014 22:03:51 GMT  
Cache-Control: public,max-age=31536000  
Content-Type: text/css  
Last-Modified: Fri, 04 Oct 2013 22:01:42 GMT  
Etag: W/"3346d-524f3ac6"  
Content-Encoding: gzip  
Content-Length: 38110  
Date: Mon, 07 Oct 2013 04:26:06 GMT  
Connection: keep-alive

-----  
(4) http://www.bloomberg.com/rapi/js\_config.js

GET /rapi/js\_config.js HTTP/1.1  
Host: www.bloomberg.com

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)  
Gecko/20100101 Firefox/21.0  
Accept: \*/\*  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
DNT: 1  
Connection: keep-alive

HTTP/1.1 200 OK  
Status: 200 OK  
X-Runtime: 24  
Content-Type: text/javascript; charset=utf-8  
Etag: "c74967659d54746bf8b22eb67163134b"  
Vary: Accept-Encoding  
Content-Encoding: gzip  
Content-Length: 1439  
Cache-Control: private, max-age=16  
Date: Mon, 07 Oct 2013 04:26:06 GMT  
Connection: keep-alive

---

(5) <http://www.bloomberg.com/favicon.ico>

GET /favicon.ico HTTP/1.1  
Host: www.bloomberg.com  
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)  
Gecko/20100101 Firefox/21.0  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
DNT: 1  
Connection: keep-alive

HTTP/1.1 200 OK  
Server: Sun-Java-System-Web-Server/7.0  
Vary: Accept-Encoding  
Content-Type: image/x-icon  
Last-Modified: Tue, 24 May 2011 23:03:08 GMT  
Etag: W/"47e-4ddc392c"  
Content-Encoding: gzip  
Content-Length: 316  
Cache-Control: max-age=161638  
Expires: Wed, 09 Oct 2013 01:20:04 GMT  
Date: Mon, 07 Oct 2013 04:26:06 GMT  
Connection: keep-alive

---

(6) [http://cdn.gottraffic.net/custom\\_contents/7bef4987c4e3a931110ba89e93b609a1.jpg](http://cdn.gottraffic.net/custom_contents/7bef4987c4e3a931110ba89e93b609a1.jpg)

GET /custom\_contents/7bef4987c4e3a931110ba89e93b609a1.jpg HTTP/1.1  
Host: cdn.gottraffic.net  
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)  
Gecko/20100101 Firefox/21.0  
Accept: image/png,image/\*;q=0.8,\*/\*;q=0.5  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
DNT: 1  
Connection: keep-alive



HTTP/1.1 200 OK  
Server: Sun-Java-System-Web-Server/7.0  
Expires: Thu, 02 Oct 2014 11:02:14 GMT  
Cache-Control: public,max-age=31536000  
Content-Type: image/jpeg  
Last-Modified: Wed, 02 Oct 2013 11:01:33 GMT  
Content-Length: 10373  
Etag: "2885-524bfd0d"  
Accept-Ranges: bytes  
Date: Mon, 07 Oct 2013 04:26:06 GMT  
Connection: keep-alive

-----  
(7) http://www.google-analyt  
analytics.com/\_\_utm.gif?utmwv=5.4.5&utms=1&utmn=2013761266&utmhn=www.bloomber  
g.com&utme=8(recommendation\_cookie\*name\*4!markets\_video\_tab\_test\*tv\_reco)9(t2  
bb03\*blp.home%3Aindex\*4!follow\*qt)11(1\*4!1\*1)&utmcs=UTF-  
8&utmsr=1440x900&utmvp=1425x735&utmsc=24-bit&utmul=en-  
us&utmje=1&utmfl=11.8%20r800&utmdt=Bloomberg%20-  
%20Business%2C%20Financial%20%26%20Economic%20News%2C%20Stock%20Quotes&utmhid  
=1044610829&utmr=-&utmp=%2F&utmht=1381119968678&utmcs=UTF-  
1&utmcc=\_\_utma%3D30057196.1607550793.1381119969.1381119969.1381119969.1%3B%2B  
\_\_utmz%3D30057196.1381119969.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmc  
md%3D(none)%3B&utmu=qQAAAAAAAAAAAAAAAAAQ~

GET  
/\_\_utm.gif?utmwv=5.4.5&utms=1&utmn=2013761266&utmhn=www.bloomberg.com&utme=8(  
recommendation\_cookie\*name\*4!markets\_video\_tab\_test\*tv\_reco)9(t2bb03\*blp.home  
%3Aindex\*4!follow\*qt)11(1\*4!1\*1)&utmcs=UTF-  
8&utmsr=1440x900&utmvp=1425x735&utmsc=24-bit&utmul=en-  
us&utmje=1&utmfl=11.8%20r800&utmdt=Bloomberg%20-  
%20Business%2C%20Financial%20%26%20Economic%20News%2C%20Stock%20Quotes&utmhid  
=1044610829&utmr=-&utmp=%2F&utmht=1381119968678&utmcs=UTF-  
1&utmcc=\_\_utma%3D30057196.1607550793.1381119969.1381119969.1381119969.1%3B%2B  
\_\_utmz%3D30057196.1381119969.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmc  
md%3D(none)%3B&utmu=qQAAAAAAAAAAAAAAAAAQ~ HTTP/1.1  
Host: www.google-analytics.com  
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)  
Gecko/20100101 Firefox/21.0  
Accept: image/png,image/\*;q=0.8,\*/\*;q=0.5  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
DNT: 1  
Connection: keep-alive

HTTP/1.1 200 OK  
Pragma: no-cache  
Expires: Wed, 19 Apr 2000 11:43:00 GMT  
Last-Modified: Wed, 21 Jan 2004 19:51:30 GMT  
X-Content-Type-Options: nosniff  
Content-Type: image/gif  
Date: Thu, 26 Sep 2013 23:07:09 GMT  
Server: Golfe2  
Content-Length: 35  
Cache-Control: private, no-cache, no-cache=Set-Cookie, proxy-revalidate  
Age: 883139  
Alternate-Protocol: 80:quic

-----  
(8) http://www.google.com/pagead/drt/ui

```
GET /pagead/drt/ui HTTP/1.1
Host: www.google.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:21.0)
Gecko/20100101 Firefox/21.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive

HTTP/1.1 302 Found
Location:
https://accounts.google.com/ServiceLogin?service=doritos&passive=true&go=true
Cache-Control: private
X-Frame-Options: ALLOWALL
Content-Type: text/html; charset=UTF-8
X-Content-Type-Options: nosniff
Date: Mon, 07 Oct 2013 04:26:09 GMT
Server: safe
Content-Length: 407
X-XSS-Protection: 1; mode=block
Alternate-Protocol: 80:quic
```

-----  
(9)  
-----

---

**Q1. What would be the GET following transaction #8?**

**Q2. How big is the HTML of the Bloomberg's home page?**

**Q3. How many redirects are listed and what are those transactions?**

**Q4. Why are transactions (3) and (4) responding with an Etag?**

**Q5. Of the servers listed, which are compressing their responses and which are not?**

**Compress:**

**Do not compress:**

**Q6. What is Google Analytics responding with?**

**Q7. Why is the content returned by transaction (6) not compressed?**

**Q8. What is the purpose of transaction (5)?**

**Q9. What content distribution network is Bloomberg using?**

**Q10. What character sets are used in these transactions?**

## **JAVASCRIPT QUESTIONS [20 PTS]**

**Consider the following XML file whose name is airbus.xml:**

```
<?xml version = "1.0"?>
<catalog>
  <title>Airbus Aircrafts</title>
  <aircraft>
    <Airbus>A380</Airbus>
    <seats>555</seats>
    <Range>15000km </Range>
    <Wingspan>79.80m</Wingspan>
    <Length>73.00m</Length>
    <Height>24.10m</Height>
    <Image>http://www-scf.usc.edu/~csci571/A380.jpg</Image>
  </aircraft>
  <aircraft>
    <Airbus>A330/A340</Airbus>
    <seats>380</seats>
    <Range>14600km</Range>
    <Wingspan>63.45m</Wingspan>
    <Length>75.30m</Length>
    <Height>17.30m</Height>
    <Image>http://www-scf.usc.edu/~csci571/A340.jpg</Image>
  </aircraft>
</catalog>
```

**Complete the JavaScript code snippet below that produces the following output:**

```
A380  
555  
A330/A340  
380
```

```
<html>  
<head><title>Airbus</title></head>  
<body>  
  
<script type="text/javascript">  
var xmlDoc = new ActiveXObject("Microsoft.XMLDOM");  
xmlDoc.async="false";  
xmlDoc.load("airbus.xml");  
  
planes=xmlDoc.getElementsByTagName("aircraft");  
planeNodeList=planes.item(0).childNodes;
```

```
</script>  
</body>  
</html>
```

## CSS QUESTIONS [10 PTS]

Below are lines extracted from the CSS sheet of espn.com. For each line answer the question. Each question is worth two (2) points.

### Q1. What does this declaration do?

```
.logo-nba-large {  
    background-image:  
url(http://a4.espncdn.com/prod/assets/teamlogos/nba/logo-nba-teams-  
large-vert.png);  
    background-position: -100px 0;  
    background-repeat: no-repeat  
}
```

**A1.**

### Q2. Give an HTML example usage of the declaration below:

```
body.page2 #header h1.phrase {  
    position: absolute;  
    top: 36px;  
    left: 189px;  
    font: 12px Helvetica, Arial, sans-serif;  
    color: #fee399;  
    padding: 2px;  
    padding-right: 10px;  
    width: 470px  
}
```

**A2.**

### Q3. Give an HTML example usage of the declaration below:

```
#header div.insider-logo a {  
    display: block;  
    height: 25px;  
    position: relative !important  
}
```

**A3.**

**Q4. Give an HTML example usage of the declaration below:**

```
.article p {  
    margin-bottom: 10px;  
    line-height: 17px;  
    color: #333  
}
```

**A4.**

**Q5. Write the CSS to create a “class” named “photo-enlarge” that applies only to the “BODY” element and a generic “id” name “header”, in such a way that when an “anchor” is inside an element with id=header, which in an HTML file with BODY with class=photo-enlarge, the following settings are modified as such:**

```
float: right;  
color: #FFF;  
font-size: 10px;  
font-family: Verdana, Arial, Helvetica, sans-serif;  
padding-right: 14px;  
margin: 8px 10px
```

**A5.**