ST2220

SINGAPORE POLYTECHNIC

2014/2015 SEMESTER ONE MID-SEMESTER TEST

DIPLOMA IN BUSINESS INFORMATION TECHNOLOGY

SECOND YEAR FULL TIME

WEB APPLICATION DEVELOPMENT

Time Allowed: 2 Hours

Instructions to Candidates

1. This paper comprises **5** questions.
2. This paper consists of **9** pages (inclusive of cover page and appendix).
3. Answer **ALL** questions.
4. All answers should be written in the answer booklet.
5. Start each question on a new page.
6. This is a close book test. Students are NOT allowed to bring any reference material.
7. Students are required to return the test papers after the test.

Student ID: _____ Name: _____

Question 1

(a) Given the PHP script in listing 1, what is the complete output?

```
<!DOCTYPE html>
<html>
    <head><title>question 1 a</title></head>
    <body>
        <?php
        $num1 = 2; $num2 = 8; $num3 = 4;

        echo "$num1 % $num2 <br />";

        $num5 = $num1 / $num3;

        echo $num3 . ' ' . $num5 . '<br />$num2 $num1';
        ?>
    </body>
</html>
```

Listing 1

(6 marks)

(b) Given the PHP script in listing 2, what is the complete output?

```
<!DOCTYPE html>
<html>
    <head><title>question 1 b</title></head>
    <body>
        <?php
        $num1 = 4; $num2 = 2; $num3 = 8; $num4 = 2;

        $num4 *= $num3;
        $num5  = --$num1 - $num2++;

        echo "$num1 <br />";
        echo "$num2 <br />";
        echo "$num3 <br />";
        echo "$num4 <br />";
        echo "$num5 <br />";

        ?>
    </body>
</html>
```

Listing 2

(10 marks)

(c) Given the PHP script in listing 3, what is the complete output?

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        $arr = array(3, 7, 4, 6);

        foreach ($arr as $index => $value) {
            echo "$value : ";

            $newValue = $index * $value;

            echo "$newValue <br />";
        }
        echo "count : " . count($arr);
        ?>
    </body>
</html>
```

Listing 3

(10 marks)

Question 2

The HTML code in listing 4 creates a webpage containing a form for the user to enter six numbers and a check number.

```
<!DOCTYPE html>
<html>
    <head>
        <title>Question 2</title>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    </head>
    <body>
        <div>
            <form action="question2.php" method="post">
                Multiplication factor to check :
                <input type="number" name="checkNum"/> <br /><br />

                value 1 <input type="number" name="numArr[]"/><br />
                value 2 <input type="number" name="numArr[]"/><br />
                value 3 <input type="number" name="numArr[]"/><br />
                value 4 <input type="number" name="numArr[]"/><br />
                value 5 <input type="number" name="numArr[]"/><br />
                value 6 <input type="number" name="numArr[]"/><br />
                <input type="submit" name="submit" value="submit" />
            </form>
        </div>
    </body>
</html>
```

Listing 4

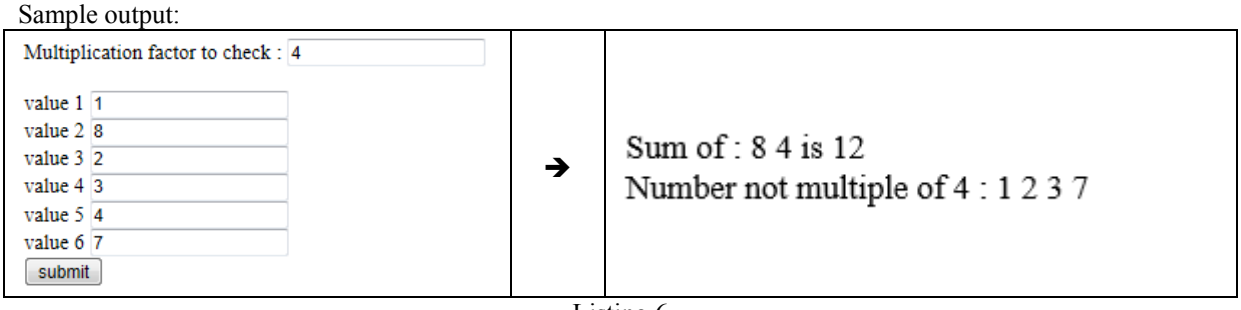
Complete the PHP script in listing 5 to handle the submission of the above HTML form. The user will enter the multiplication factor number for checking, the script will check which of the entered numbers is a multiple of that factor number and sum them up. The script will also indicate which numbers are not a multiple of the check number (Assume the user will always enter numbers which are more than zero). You may refer to listing 6 for a sample output.

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        $numArr = (a)_____;
        $check  = (b)_____;
        $total  = (c)_____;

        $resultStr1 = "Sum of : ";

        $resultStr2 = "Number not multiple of $check : ";

        for ( (d)_____ ) {

            if ( (e)_____ ) {

                (f)_____;

                $total += (g)_____;

            } else {

                (h)_____;

            }
        }

        $resultStr1 .= " is $total";

        echo "(i)_____ <br />";
        echo "(j)_____ <br />";
        ?>
    </body>
</html>
```

Listing 5

(25 marks)

Sample output:



Listing 6

Question 3

Given the following PHP script in listing 7:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        $productArr = array (
               "ac001" => array("chair","50.00","available"),
               "ct002" => array("coffee table", "120.00", "out of stock"),
               "sd007" => array("sideboard", "750.00", "available"));

        if (isset($_GET["id"])) {
            if (isset($productArr[$_GET["id"]])) {
                $product = $productArr[$_GET["id"]];
                echo $product[2] . " " . $product[0] . " at $" .
                     $product[1];
            } else {
                echo "NIL";
            }
        } else {
            foreach ($productArr as $k => $v) {
                echo "$k is $v[2]<br>";
            }
        }
        ?>
    </body>
</html>
```

Listing 7

Assume the URL to access the PHP script from question 3 is

`http://localhost/question03.php`

What is the output if the script is run via the following URL configuration?

(a)

```
http://localhost/question03.php?id=sd007
```

(6 marks)

(b)

```
http://localhost/question03.php
```

(6 marks)

(c)

```
http://localhost/question03.php?id=AC001
```

(3 marks)

Question 4

Complete the following PHP script in listing 8 to produce the output as shown in the sample output in listing 9.

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>question 4</title>
    </head>
    <body>
        <?php
        function findMinMaxScore($arr, $op) {
            $result = $arr[0];

            for ($i=1; $i<count($arr); $i++) {
                if ($op == 0) {
                    if ($arr[$i] < $result) {
                        $result = $arr[$i];
                    }
                } else if ($op == 1) {
                    if ($arr[$i] > $result) {
                        $result = $arr[$i];
                    }
                }
            }
            return $result;
        }

        $sArr = array (array(3, 6, 9, 2),
                       array(4, 7, 8),
                       array(3, 1, 1, 4, 5, 3));

        $minValue = 99;
        $maxValue = 0;

        for ( (a)_____________________________ ) {

            (b)_________ = findMinMaxScore( (c)_______ , (d)________ );

            if ($maxValue < $currMax) {
                $maxValue = (e)________ ;
            }

            (f)_________ = findMinMaxScore( (g)_______ , (h)________ );

            if ($minValue > $currMin) {
                $minValue = (i)________ ;
            }
        }

        echo "Highest number in the 2D array is $maxValue <br />";
        echo "Lowest number in the 2D array is $minValue";
        ?>
    </body>
</html>
```

Listing 8

(14 marks)

Sample Output:

```
Highest number in the 2D array is 9
Lowest number in the 2D array is 1
```

Listing 9

Question 5

Given "data.txt" in listing 10 which contains information as shown below:

```
I love your product.
Well priced.
```

Listing 10

The HTML code in listing 11 creates a webpage containing a form for the user to enter feedback.

```
<html>
    <head>
        <title>Question 5</title>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    </head>
    <body>
        <div>
            <form action="question5.php" method="post">
                Submit your feedback:<br />
                <textarea name="feedback" cols="30" rows="5"></textarea>
                <br />
                <input type="submit" name="submit" value="submit" />
            </form>
        </div>
    </body>
</html>
```

Listing 11

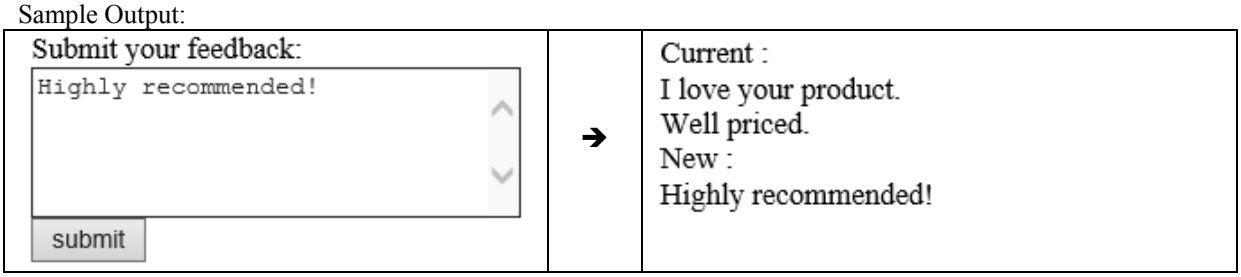
Complete the PHP script in listing 12 which reads from the "data.txt" file and display the information. It will then append the feedback read from the submitted form to the end of the file. You may refer to listing 13 and listing 14 for a sample output.

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        $feedback = (a)_____;

        $infile = (b)_____ ( (c)_____ , (d)_____ ) or
                  exit("Unable to open file!");

        echo "Current : <br />";

        while ( (e)_____ ) {

            echo (f)_____ . "<br />";

        }

        echo "New : <br />";
        echo (g)_____;

        (h)_____; //append new feedback to file

        (i)_____; // close file
        ?>
    </body>
</html>
```

Listing 12

(20 marks)

Sample Output:



Listing 13

The "data.txt" in listing 14 which contains information as shown below after the new feedback is appended:

```
I love your product.
Well priced.
Highly recommended!
```

Listing 14

- End of Paper -

Appendix

Operators

Operator	Name	Example	Result
x + y	Addition	2 + 3	5
x - y	Subtraction	7 - 4	3
x * y	Multiplication	5 * 2	10
x / y	Division	9 / 3	3
x % y	Modulus	5 % 2	1
		10 % 8	2
		3 % 10	3
- x	Negation	-2	-2
a . b	Concatenation	"Ya" . "Ho"	YaHo

Assignment	Same as...	Description
x = y	x = y	assign y to x
x += y	x = x + y	Increase x by y
x -= y	x = x - y	Subtract x by y
x *= y	x = x * y	Multiply x by y
x /= y	x = x / y	Divide x by y
x %= y	x = x % y	Modulus x by y
a .= b	a = a . b	Concatenate two strings

Operator	Name	Description
++x	Pre-increment	Increments x by one, then returns x
x++	Post-increment	Returns x, then increments x by one
--x	Pre-decrement	Decrements x by one, then returns x
x--	Post-decrement	Returns x, then decrements x by one

Function

Function	Description
count()	is used to return the length (the number of elements) of an array.
explode()	is used to split a string parameter on boundaries formed by the delimiter into an array.
fopen()	is used to open files in PHP.
fclose()	is used to close an open file.
feof()	checks if the "end-of-file" (EOF) has been reached.
fgets()	is used to read a single line from a file.
fwrite()	Is used to write to a file.

File may be opened in one of the following modes

Modes	Description
r	Read only. Starts at the beginning of the file
r+	Read/Write. Starts at the beginning of the file
w	Write only. Opens and clears the contents of file; or creates a new file if it doesn't exist
w+	Read/Write. Opens and clears the contents of file; or creates a new file if it doesn't exist
a	Append. Opens and writes to the end of the file or creates a new file if it doesn't exist
a+	Read/Append. Preserves file content by writing to the end of the file
x	Write only. Creates a new file. Returns FALSE and an error if file already exists
x+	Read/Write. Creates a new file. Returns FALSE and an error if file already exists

- End of Appendix -