

SINGAPORE POLYTECHNIC

2014/2015 SEMESTER TWO 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 **12** pages (inclusive of cover page and appendix).
3. Answer **ALL** questions.
4. All answers should be written in the answer booklet.
5. This is a close book test. Students are NOT allowed to bring any reference material.
6. 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><meta charset="UTF-8"><title>Q1a</title></head>
  <body>
    <?php
      $num1 = 7; $num2 = 2; $num3 = true; $num4 = false;

      echo ($num1 * $num2) . "<br />";
      echo "$num1 - $num2 <br />";

      echo '$num3 and $num4 <br />';
      echo ($num3 && $num4) ? "A" : "B" . ' <br />';
    ?>
  </body>
</html>
```

Listing 1

(8 marks)

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

```
<!DOCTYPE html>
<html>
  <head><meta charset="UTF-8"><title>Q1b</title></head>
  <body>
    <?php
      $a = true; $b = false; $c = 4;
      $d = $c % 8 == 4;

      if (!($a && !$b)) {
        echo "A<br />";
      } else {
        echo "B<br />";
      }

      if ($d) {
        echo "C<br />";
      } else if ($a) {
        echo "D<br />";
      } else {
        echo "E<br />";
      }

      echo "F<br />";
    ?>
  </body>
</html>
```

Listing 2

(6 marks)

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

```
<!DOCTYPE html>
<html>
  <head><meta charset="UTF-8"><title>Q1c</title></head>
  <body>
    <?php
      $arr = ["A" => 8, "B" => 7, "C" => 6, "D" => 5];

      $strMsg1 = "";
      $strMsg2 = 0;
      foreach ($arr as $x => $y) {
        echo "$y$x<br />";

        $strMsg1 .= $x;
        $strMsg2 += $y;
      }
      echo "$strMsg1 <br />";
      echo "$strMsg2 <br />";
    ?>
  </body>
</html>
```

Listing 3

(8 marks)

Question 2

The HTML code in listing 4 creates a webpage containing a form for the user to enter seven numbers.

```
<!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">
        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 />
        value 7 <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 script will first sort the numbers in ascending order, then print the numbers and determine whether the numbers entered are unique or not, i.e. no duplication of numbers entered. The script will construct an array to hold the numbers with all duplicates removed. Refer to listing 6 for the sample output. (Note: once an array of numbers is sorted, duplicates can be easily detected as duplicate numbers will appear side by side.)

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Question 2</title>
  </head>
  <body>
    <?php
      $givenArr = (a)_____ ; // get the array of numbers from the form
      $unique    = true;

      (b)_____ ; // sort the array $givenArr in ascending order

      $newArr[0] = (c)_____ ;
      // assign the first number from array $givenArr to the first
      // index of a array $newArr

      echo "Given Array sorted in ascending order:<br />" .
      (d)_____ . " ";

      for ( (e)_____ ) {
        // go through the rest of the array $givenArr starting from index
        // position 1

        echo (f)_____ . " ";

        if ( (g)_____ ) {
          // check if current number is the same as the number in the
          // index position before it.

          $unique = (h)_____ ; // update variable $unique
        } else {
          (i)_____ = $givenArr[$i];
          // append current number to array $newArr
        }
      }
      echo "<br />";

      if ( (j)_____ ) {
        echo "The list of numbers is unique.";
      }
      else {
        echo "The list of numbers is not unique.<br />";
        echo "New list with duplicate removed:<br/>";
        foreach ( (k)_____ ) {
          echo "$value ";
        }
      }
    ?>
  </body>
</html>
```

Listing 5

(25 marks)

Sample output:

<div><div>value 1</div><div>5</div></div> <div><div>value 2</div><div>9</div></div> <div><div>value 3</div><div>6</div></div> <div><div>value 4</div><div>1</div></div> <div><div>value 5</div><div>3</div></div> <div><div>value 6</div><div>11</div></div> <div><div>value 7</div><div>4</div></div> <div><div>submit</div></div>	<div>→</div>	<div>Given Array sorted in ascending order: 1 3 4 5 6 9 11 The list of numbers is unique.</div>
<div><div>value 1</div><div>5</div></div> <div><div>value 2</div><div>9</div></div> <div><div>value 3</div><div>5</div></div> <div><div>value 4</div><div>1</div></div> <div><div>value 5</div><div>3</div></div> <div><div>value 6</div><div>11</div></div> <div><div>value 7</div><div>9</div></div> <div><div>submit</div></div>	<div>→</div>	<div>Given Array sorted in ascending order: 1 3 5 5 9 9 11 The list of numbers is not unique. New list with duplicate removed: 1 3 5 9 11</div>

Listing 6

Question 3

Given the following PHP script in listing 7:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Question 3</title>
  </head>
  <body>
    <?php
      $libraryArr = [
        "isbn978312" => array( "Social media marketing",
                              "Tuten Tracy",
                              "Main Lending",
                              "available"),
        "isbn979634" => array( "Contemporary marketing",
                              "David L. Kurtz",
                              "Main Lending",
                              "available"),
        "mcp4041"    => array( "Black swans and wormhole wizards",
                              "Joe Satriani",
                              "Hilltop media",
                              "on loan"),
        "mcp4002"    => array( "Closer to the truth",
                              "Cher",
                              "Main media",
                              "available")
      ];

      if (isset( $_GET["callno"])) {

        if (isset($libraryArr[$_GET["callno"]])) {

          if ($libraryArr[$_GET["callno"]][3] != "on loan") {

            $item = $libraryArr[$_GET["callno"]];

            echo $item[1] . "'s " . $item[0] . " at " . $item[2];

          } else {

            echo "Item on loan.";

          }

        } else {

          echo "Invalid item.";

        }

      } else {

        foreach ($libraryArr as $k => $v) {

          echo "$k is $v[0], by $v[1], $v[3] <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?callno=mcp4002`

(3 marks)

(b)

`http://localhost/question03.php`

(5 marks)

(c)

`http://localhost/question03.php?callno=isbn979134`

(2 marks)

(d)

`http://localhost/question03.php?call_On=isbn978312`

(5 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 charset="UTF-8"><title>Q4</title></head>
  <body style="font-family: Courier">
    <?php
      // this function compute and return the average of a given array.
      function getAverage($arr) {
        $total = 0;
        foreach ($arr as $num) {
          (a)_____ ;
        }
        return (b)_____ ;
      }

      $MSTScoreArr = ["Science" => [65, 55, 98, 76, 35],
                     "Math"      => [58, 78, 60, 89, 77],
                     "English"   => [70, 65, 82, 55, 76],
                     "Chinese"   => [50, 55, 57, 67, 75]];

      $averageArr = array();

      foreach ((c)_____ as $subject => $scores) {
        $averageArr[$subject] = (d)_____ ;
        echo "Average score for $subject : (e)_____ <br/>";
      }

      (f)_____ ;

      echo "<p>Subject with the lowest average : " .
        key($averageArr) . " at " . current($averageArr) . " <br/>";

      (g)_____ ;

      echo "Subject with the highest average : " .
        key($averageArr) . " at " . current($averageArr) . " <br/></p>";
    ?>
  </body>
</html>
```

Listing 8

(14 marks)

Sample Output:

```
Average score for Science : 65.8
Average score for Math : 72.4
Average score for English : 69.6
Average score for Chinese : 60.8
```

```
Subject with the lowest average : Chinese at 60.8
Subject with the highest average : Math at 72.4
```



Listing 9

Question 5

The file “album.dat” in listing 10 contains music album information as shown below. Each row shows the title of the album, the artist name followed by the rating of the album, which is separated by the ‘|’ character. The highest rating an album can obtain is 5.

```
Shake It Off|Taylor Swift|4
Sonic Highways|Foo Fighters|5
The Heart Wants What It Wants|Selena Gomez|3
```

Listing 10

Complete the PHP script in listing 11 which reads from the “album.dat” file and displays the information. You may refer to listing 12 for the sample output. Note the image filename for the image  is goodstar.jpg and the filename for the image  is badstar.jpg. Assume, both image files are located in the same folder as the PHP file.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Question 5</title>
  </head>
  <body>
    <?php
      // open file for reading, exit the script if file file is unable
      // to open.
      $infile = (a)_____ or (b)_____ ("Unable to open file!");

      while ((c)_____) {
        $inputStr = (d)_____;

        // invoke the explode function to split the input string into
        // an array.
        $albumInfoArr = (e)_____;

        echo "<p>";
        echo "Album Title: " . (f)_____ . "<br />";
        echo "Artist: " . (g)_____ . "<br />";

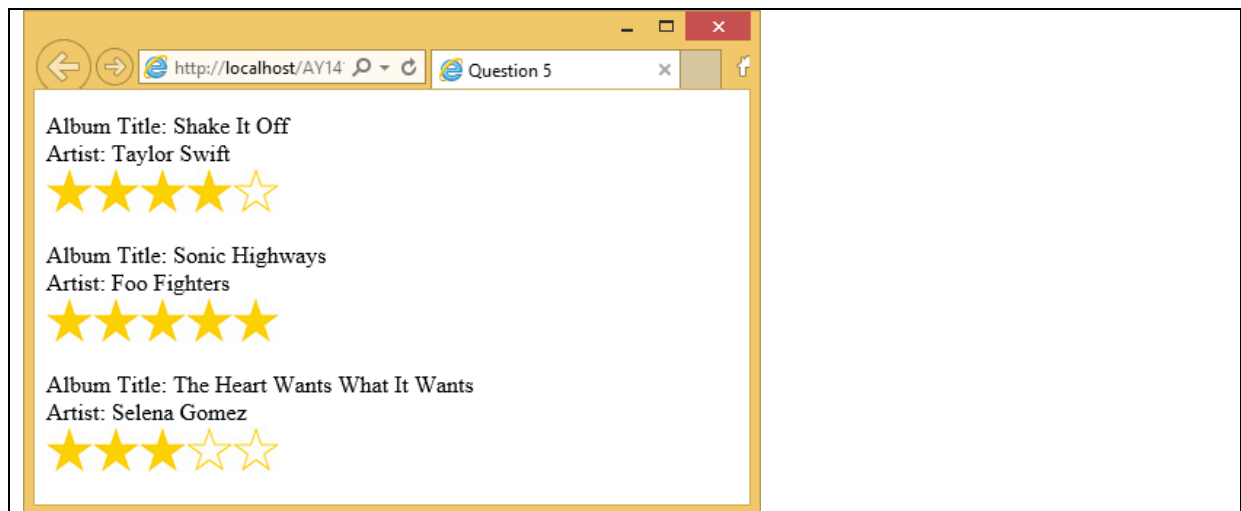
        $rating = (int) (h)_____;

        // display the stars rating
        for ($i=0; (i)_____; $i++) {
          echo "<img src='(j)_____' />";
        }
        for ((k)_____; $i<5; $i++) {
          echo "<img src='(l)_____' />";
        }
        echo "</p>";
      }
      (m)_____; // close the file
    ?>
  </body>
</html>
```

Listing 11

(24 marks)

Sample Output:



Listing 12

- End of Paper -

Appendix

Function	Description
count()	is used to return the length (the number of elements) of an array. // Example \$a[0] = 1; \$a[1] = 3; \$a[2] = 5; \$result = count(\$a); // \$result is 3
sort()	This function sorts an array. Elements will be arranged from lowest to highest when this function has completed. \$xArr = array(5, 6, 2, 3); sort(\$xArr); Resulting array will be: xArr [0] contains 2 xArr [1] contains 3 xArr [2] contains 5 xArr [3] contains 6
asort()	Elements will be arranged from lowest to highest. This function sorts a given array such that array indices maintain their correlation with the array elements they are associated with. This is used mainly when sorting associative arrays where the actual element order is significant.
arsort()	Elements will be arranged from highest to lowest. This function sorts a given array such that array indices maintain their correlation with the array elements they are associated with. This is used mainly when sorting associative arrays where the actual element order is significant.
key()	Fetch a key from an array of the current array position.
current()	Return the current element in an array.
explode()	is used to split a string parameter on boundaries formed by the delimiter into an array. // Example \$pizza = "piece1*piece2*piece3*piece4*piece5*piece6"; \$pieces = explode("*", \$pizza); echo \$pieces[0]; // result is "piece1" echo \$pieces[1]; // result is "piece2"
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.

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 -