Foss Lab

Rwithik Manoj
Roll No. 53
TVE17CS054

May 16, 2019

Problem 1

Problem Statement

Write a bash script to find the binary equivalent of a given number.

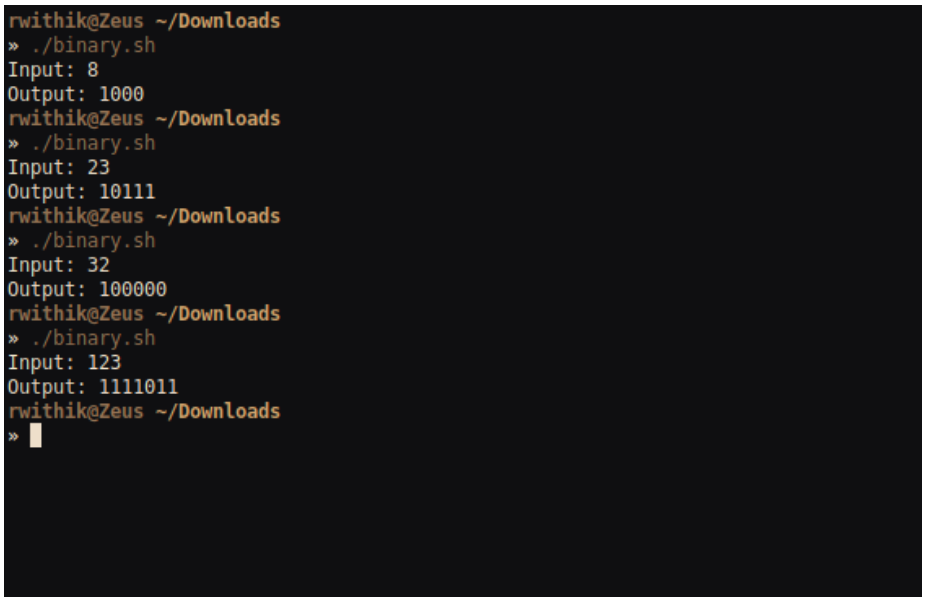
Theory

A decimal numbers binary equvalent can be found by repeatedlt prepending the remainder after dividing the number by 2.

Implementation

The script reads the decimal number into a variable a and declares the variable b as an empty string. Then it loops as long as a is greater than 0. In each iteration, it prepend the remainder after dividing a by 2 and then divides a by 2 using floor division. This is more effecient than appending the remainder and then reversing the string with another loop.

Test Runs



Code

```
#!/bin/bash

echo -n "Input: "
read a      #Read the input

b=""

while [[ $a -gt 0 ]]
do
    b="$(( $a%2 ))"$b     # Prepend the remainder to b
    a=$(( $a/2 ))         # Floor division of a by 2
done

echo "Output: $b"
```

Problem 2

Problem Statement

Given a file containing the marks obtained by students for 3 subjects in an exam. In order to pass, student should score at least 50 marks in every subject. The file has one record(line) for each student in the following format:
roll_number subject1 subject2 subject3

Theory

Read the file with awk. Check if all the marks of a roll number are at least 50. If it is, print "pass", else print "fail".

Implementation

The implementation is simple. An if condition checks if all the marks are greater than 50 or not and print the appropriate result.

Test Runs

```
rwithik@Zeus Scripts/Exam (master *%)
» cat entries
TVE14CS021 88 49 69
rwithik@Zeus Scripts/Exam (master *%)
» awk -f marks entries
TVE14CS021 fail
rwithik@Zeus Scripts/Exam (master *%)
» cat entries
TVE14CS021 88 49 69
TVE14CS022 38 59 29
TVE14CS023 48 28 24
TVE14CS024 82 68 63
TVE14CS025 83 59 76
rwithik@Zeus Scripts/Exam (master *%)
» awk -f marks entries
TVE14CS021 fail
TVE14CS022 fail
TVE14CS023 fail
TVE14CS024 pass
TVE14CS025 pass
rwithik@Zeus Scripts/Exam (master *%)
»
```

Code

```
#!/usr/bin/awk -f

{
if ($2 > 49 && $3 > 49 && $4 > 49)
        print $1, "pass";
else
        print $1, "fail";
}
```

Problem 3

Problem Statement

Implement PHP application that asks a random question(from a given set) to the user and evaluates if the user's answer is correct.

Theory

There is a file with all the questions and answers. Read all the questions and answers from the file and choose a random question to display. Check if the answer submitted by the user is correct and display the corrosponding message.

Implementation

The questions.php file checks if the session item storing the questions or the one storing the answers is empty. It any of them are empty, then it read the questions and answers from the file and inserts them into the session items.

The form sends a post request to the same file where the answer entered by the user is compared to the answer entered by the user and the correct answer. If they match, the user is redirected to the success.php file after setting a session variable named success.

In the success.php file, the success session variable is checked and if it is set, the message is displayed. Otherwise the user is redirected to the questions.php file.

Code

questions.php

```php
<?php

session_start();

$error = 0;
$_SESSION['success'] = 0;

if (isset($_POST['submit'])){
    if(empty($_POST['answer'])){
```

```
        $error = -1;
    }
    elseif(strcasecmp($_SESSION['answer'], $_POST['answer'])){
        $_SESSION['success'] = 1;
        header('location: success.php');
    }
    else{
        $error = 1;
    }
}

if(!isset($_SESSION['questions']) || $_SESSION['answers']){

    $f = fopen("myfile.txt","r");
      $questions = array();
      $answers = array();

      while(! feof($f))  {
          // Read from file to the questions and answers arrays.
        $result = fgets($f);
        array_push($questions, $result);
        $result = fgets($f);
        array_push($answers, $result);
        $blank = fgets($f);
      }
      fclose($f);
      // Store the arrays in session variables.
      $_SESSION['questions'] = $questions;
      $_SESSION['answers'] = $answers;
}

?>

<html>
<head><title>Questions</title></head>
<body style="text-align: center; margin-top: 30vh">
    <?php
        // Select a random question number
        $no = rand(1, 28);
        echo $no;
        // Store the correct answer in a session variable.
        $_SESSION['answer'] = $answers[$no];
        echo $questions[$no];
    ?>
    <form method="post" action="questions.php" >
        Answer: <input type="text" name="answer" placeholder="Enter the Answer">
        <br>
        <br>
        <input type="submit" value="Submit" name="submit">
    </form>
```

```
    <?php
        // Display the errors
        if($error == -1){
            echo "<p style='color: red'>Enter an answer!</p>";
        }
        elseif($error == 1){
            echo "<p style='color: red'>Wrong answer!</p>";
        }
    ?>

</body>

</html>
```

success.php

```
<?php
    // Check if the session variable is set
    session_start();
    if($_SESSION['success'] == 1){
        $_SESSION['success'] = null;
        echo "Correct answer";
    }
    else{
    header('location: questions.php');
    }
?>
```