VILNIUS UNIVERSITY
ŠIAULIAI ACADEMY

BACHELOR PROGRAMME SOFTWARE ENGINEERING

Object Oriented Programming

Practical 1 (One).

Student: Sunday Emmanuel Sanni

Lecturer: Prof. Donatas Dervinis

Šiauliai,

15/02/2024

1. **Create PHP a program to calculate the mathematical function 3x + 1:** The screenshot below is a php program that accepts positive non-zero input from the user and generates a Collatz sequence of thee number. It checks for negative input on line 9, and the block of else-if from line 10 to line 15 deals with the case of the number 1.

   The number to generate a Collatz sequence for can be inputted through an html form. After the sequence is generated, the number inputted, the maximum number in the sequence, and the number of iterations or stopping time of the sequence is printed out.

singleProgram.php ● | collatzConjecture.php | function2.php

singleProgram.php

```php
<?php
$count = 0;
$highest = $highest1 = $highest2 = 0;
if (empty($_POST["variable"])) {
} else {
    $input = $_POST['variable'];
    $number = $input;
    if ($input < 0) {
        echo "Invalid Input! Try Again!";
    } elseif ($input == 1) {
        echo $input;
        echo "<br>";
        echo "Highest Value = $highest";
        echo "<br>";
        echo "Total iteration = $count";
    } else {
        echo "$input \t";
        $highest = $input;
        while ($input != 1) {
            if ($input % 2 == 0) {
                $input = $input / 2;
            } else {
                $input = ((3 * $input) + 1);
            }
            if ($highest > $input) {
            } else {
                $highest = $input;
            }
            echo "$input \t";
            $count = $count + 1;
        }
        echo "$number <br>";
        echo "$highest <br>";
        echo "$count <br>";
    }
}
?>
```

Figure 1: A screenshot of PHP program that generates Collatz sequence for a given positve number

2. **First test the program with a single number and print all the values, find max value (Highest number) and count of iterations (Total stopping time):**

   The 2 images below show the HTML page that is used to accept an input from the user. The images below are examples of the sequence and results generated from the numbers 30 and 50

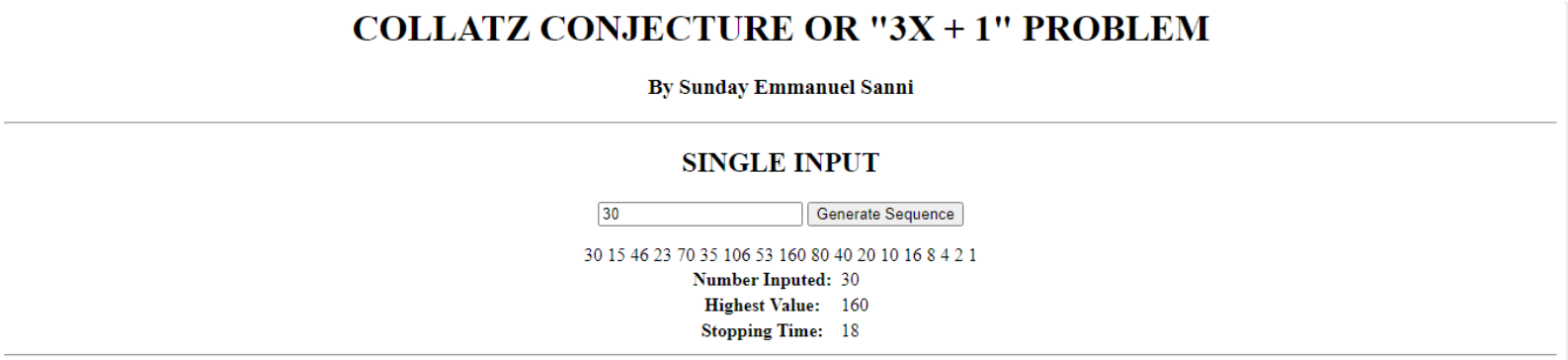

Figure 2: Collatz Sequence, with highest value and Stopping Time for the number 30.

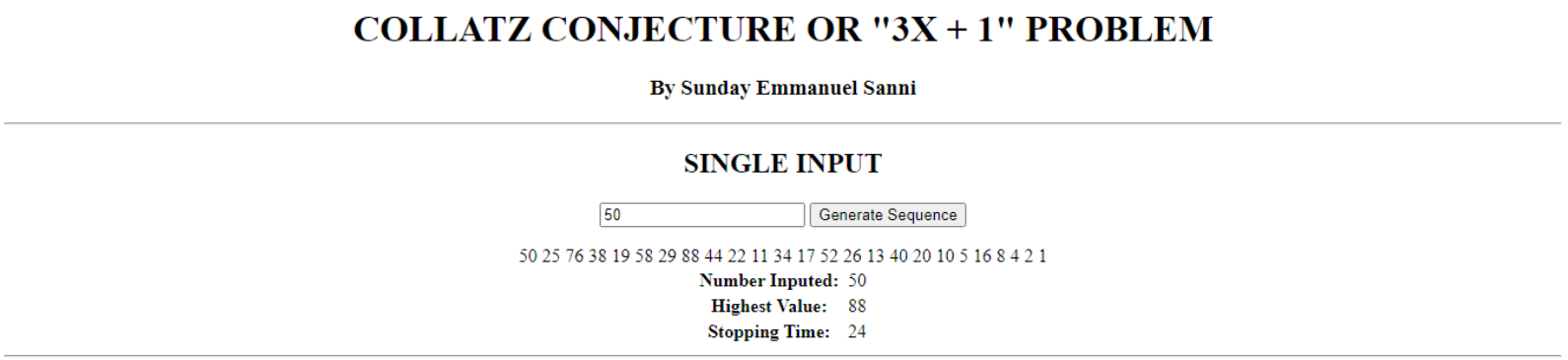

Figure 3: Collatz Sequence, with highest value and Stopping Time for the number 50.

3. **Create extra function for calculating functions values (max value and count of iterations) of numbers range (e.g., 10 to 10^6):**

   The image below is a function that takes a range of positive values. It performs an operation on each number in the specified range and generates a Collatz sequence for each number. The sequence generated for each number is not printed. The sequence for each number undergoes a process that brings out the highest number in the sequence and count the number of iteration. For each number in the sequence, highest value obtained and number of iteration is stored in a 2 dimensional array. The two dimensional array is returned by the function.

   This returned array is used by another PHP code in another file. The PHP code that receives this array prints out the Minimum and Maximum iteration, the number that generated it, and the highest value of the sequence generated by such number.

singleProgram.php | collatzConjecture.php | function2.php

function2.php > rangeOperation

```php
<?php
function rangeOperation($lowerBound, $upperBound)
{
    $highest = $number = 0;
    $lowerBound = $_POST['lowerBound'];
    $upperBound = $_POST['upperBound'];

    if ($upperBound < $lowerBound) {
        echo "Upper Range Value cannot be lesser than Lower Range Value! Try Again!";
        return NULL;
    } elseif ($lowerBound == $upperBound) {
        echo "Range of Value is the Same, use the single value Collatz Sequence Generator!";
        return NULL;
    } else {
        $dataArray = array();
        for ($i = $lowerBound; $i <= $upperBound; $i++) {
            $count = 0;
            $highest = $i;
            $number = $i;
            while ($number != 1) {
                if ($number % 2 == 0)
                    $number = $number / 2;
                else
                    $number = ((3 * $number) + 1);
                if ($number > $highest)
                    $highest = $number;
                $count = $count + 1;
            }
            array_push($dataArray, array($i, $highest, $count));
        }
        return $dataArray;
    }
```

Figure 4: Function to check highest values and number of iterations of a range of numbers, then store it in an array, before returning the array of numbers

4. **The HTML Forms must be created and prompted for range values (start and finish). As results store in array: numbers, maximum (highest) values and the number of iterations of them:**

   This HTML page below is used to create a form to accept inputs from the range of values. It is designed using the POST method. The result ca be seen in figure 5 below. A table containing the range of numbers entered is generated, with the minimum and maximum vales generated in the sequence, and also the number of iteration for each sequence. Also, two other tables are generated that shows the maximum and minimum iterations, the numbers that have these iterations, and the maximum values generated by the Collatz sequence of these numbers.

```php
<?php
include 'function2.php';
?>
<html>

<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>COLLATZ CONJECTURE</title>
</head>

<body>
    <h1 align='center'>COLLATZ CONJECTURE OR "3X + 1" PROBLEM</h1>
    <h3 align='center'> By Sunday Emmanuel Sanni</h3>
    <hr />
    <div align='center'>
        <h2 align='center'>RANGE OF VALUES</h2>
        <form method="POST" action="./collatzConjecture.php">
            <input type="number" name="lowerBound" placeholder="Minimum Number" />
            <input type="number" name="upperBound" placeholder="Maximum Number" />
            <button>Show Results</button>
        </form>
        <?php
```

Figure 5: HTML code that is used to set up the form for receiving non-zero positive range of numbers

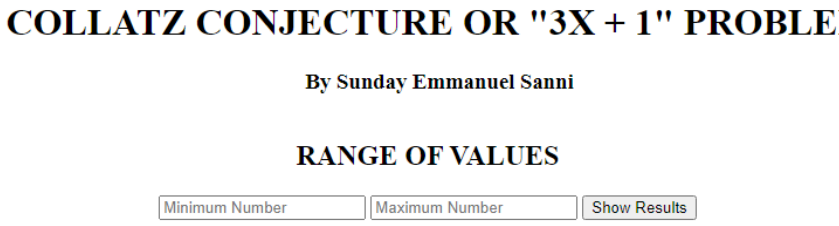

Figure 6: Display of HTML page of figure 5 in a web browser

5. **Find the numbers with have max and min iterations. Print both (2) number, count of iteration and highest values:**

   The image below is a sample print out of the implementation of the code in figure 5. It shows how in the range of numbers 20 to 30, the highest values and stopping time for each number in the range is generated for the given range of values.

RANGE OF VALUES

20 | 30 | Show Results

Number	Highest Value	Stopping Time
20	20	7
21	64	7
22	52	15
23	160	15
24	24	10
25	88	23
26	40	10
27	9232	111
28	52	18
29	88	18
30	160	18

MINIMUM STOPPING TIME		
Number	**Highest Value**	**Stopping Time**
20	20	7

MAXIMUM STOPPING TIME		
Number	**Highest Value**	**Stopping Time**
27	9232	111