<?php

class Calculation {

private $number;

private $maxValue;

private $iterations;

public function \_\_construct($number, $maxValue, $iterations) {

$this->number = $number;

$this->maxValue = $maxValue;

$this->iterations = $iterations;

}

public function getNumber() {

return $this->number;

}

public function getMaxValue() {

return $this->maxValue;

}

public function getIterations() {

return $this->iterations;

}

}

class CalculationManager {

public static function nextCalculation($x) {

return $x % 2 == 0 ? $x / 2 : 3 \* $x + 1;

}

public static function generateProgression($start, $length) {

$progression = [];

$currentValue = $start;

for ($i = 0; $i < $length; $i++) {

$number = $currentValue;

$maxValue = $currentValue;

$iterations = 0;

while ($currentValue != 1) {

$currentValue = self::nextCalculation($currentValue);

$maxValue = max($maxValue, $currentValue);

$iterations++;

}

$progression[] = new Calculation($number, $maxValue, $iterations);

$currentValue = $number + 1; // Next number in the sequence

}

return $progression;

}

}

// Example usage:

$progression = CalculationManager::generateProgression(1, 10);

foreach ($progression as $calculation) {

echo "Number: " . $calculation->getNumber() . ", Max Value: " . $calculation->getMaxValue() . ", Iterations: " . $calculation->getIterations() . "\n";

}

?>