<?php

// find the minimum of two numbers

function findMinOfTwo($num1, $num2) {

return min($num1, $num2);

}

echo findMinOfTwo(5, 9) . "<br>";

// find the minimum of three numbers

function findMinOfThree($num1, $num2, $num3) {

return min($num1, $num2, $num3);

}

echo findMinOfThree(5, 9, 3) . "<br>";

// find the minimum of two or three numbers

function findMin($num1, $num2, $num3 = null) {

if ($num3 === null) {

return min($num1, $num2);

}

else {

return min($num1, $num2, $num3);

}

}

echo findMin(7, 4,) . "<br>";

// take in two arguments, return 10 in the a-th place of n

function findNum($n, $a) {

$str = strval($n);

return $str[-$a - 1];

}

echo findNum(1541, 2) . "<br>";

echo findNum(415, 1) . "<br>";

echo findNum(7474, 0) . "<br>";

// recursive even or odd function

function isEven($num) {

if ($num == 0) {

return "Even";

}

else if ($num == 1) {

return "Odd";

}

else if ($num < 0) {

return isEven(-$num);

}

else {

return isEven($num - 2);

}

}

echo isEven(50) . "<br>";

echo isEven(75) . "<br>";

echo isEven(-1) . "<br>";

// nth term of the series

function sequence1($num) {

$ans = 2;

for ($i = 1; $i < $num; $i++) {

$ans += (4 \* $i);

}

return $ans;

}

echo sequence1(6) . "<br>";

// nth term of the series

function sequence2($num) {

$ans = 1;

for ($i = 1; $i <= $num; $i++) {

$ans \*= $i;

}

return $ans;

}

echo sequence2(6) . "<br>";

// nth term of the Fibonacci sequence

// nth term of the series

function fibonacci($num) {

$firstNum = 0;

$secondNum = 1;

$result = 0;

for ($i = 1; $i < $num; $i++) {

$result = $firstNum;

$firstNum = $secondNum;

$secondNum = $result + $firstNum;

}

return $secondNum;

}

echo fibonacci(7) . "<br>";

// HCF of two numbers

function hcf($num1, $num2) {

$ans = 0;

for ($i = 1; $i < $num1 && $i < $num2; $i++) {

if ($num1 % $i === 0 && $num2 % $i === 0) {

$ans = $i;

}

}

return $ans;

}

echo hcf(12, 16) . "<br>";

// LCM of two numbers

function lcm($num1, $num2) {

$ans = 0;

for ($i = 1; ; $i++) {

$ans = $num1 \* $i;

if ($ans % $num2 === 0) {

break;

}

}

return $ans;

}

echo lcm(10, 5) . "<br>";

// multiplication table

$x = readline("Enter your factor: ");

$a = readline("Enter your starting multiplier: ");

$b = readline("Enter your ending multiplier: ");

function multiply($x, $a, $b) {

for ($i = $a; $i <= $b; $i++) {

echo "$x \* $i = " . $x \* $i . "\n";

}

}

multiply($x, $a, $b);

// multiplication tables

function multiMultiply() {

$x = readline("Enter your starting factor: ");

$y = readline("Enter your ending factor: ");

$a = readline("Enter your starting multiplier: ");

$b = readline("Enter your ending multiplier: ");

for ($i = $x; $i <= $y; $i++) {

multiply($i, $a, $b);

echo "\n\n";

}

}

multiMultiply(5, 10, 1, 10);

// Bean counting pt 1

function countBs($str) {

$bCount = 0;

for ($i = 0; $i < strlen($str); $i++) {

if ($str[$i] === 'B') {

$bCount++;

}

}

return $bCount;

}

// Bean counting pt 2

function countChar($str, $letter) {

$letterCount = 0;

for ($i = 0; $i < strlen($str); $i++) {

if ($str[$i] === $letter) {

$letterCount++;

}

}

return $letterCount;

}

// Leetcode #7

function reverse($x) {

$str = strval($x);

$str = strrev($str);

$num = intval($str);

if ($x === 0) {

return 0;

}

else if ($x > 2147483647 || $x < -2147483648 || $num > 2147483647 || $num < -2147483648) {

return 0;

}

else if ($x < 0) {

return 0 - $num;

}

else {

return $num;

}

}

// Leetcode 1688

function numberOfMatches($n) {

$matchCount = 0;

while ($n > 1) {

// if even, each team plays another team

if ($n % 2 == 0) {

// n / 2 matches are played, half move to next round

$matchCount += $n / 2;

$n = $n / 2;

}

// if odd, one team moves on and the rest play each other

else {

// (n - 1) / 2 + 1 teams move on

$matchCount += ($n - 1) / 2;

$n = ($n - 1) / 2 + 1;

}

}

// return matches played until winner is decided

return $matchCount;

}

// Leetcode #1323

function maximum69Number ($num) {

$newNum = 0;

$str = strval($num);

for ($i = 0; $i < strlen($str); $i++) {

$newStr = $str;

if ($str[$i] == '6') {

$newStr[$i] = '9';

}

else if ($str[$i] == '9') {

$newStr[$i] == '6';

}

if (intval($newStr) > $newNum) {

$newNum = intval($newStr);

}

}

return $newNum;

}

?>