PHP Programming Tasks Report

Nested Conditional Logic & String Manipulation Exercises

Prepared by: Automated Codex Agent

This document presents two PHP programming tasks, detailing their aims, problem statements, constraints, procedures, implementations, sample outputs, and conclusions.

Task 1: Largest of Three Numbers Using Nested If

AIM

Determine the largest value among three given integers using nested conditional statements.

Problem Statement

Build a PHP CLI program that accepts three predefined integers and evaluates them through nested if statements to identify the maximum.

Constraints

- The logic must rely on nested if statements rather than built-in max functions.
- The program should clearly display the input numbers and the computed largest number.

Procedure

1. Define an array containing three integer values.
2. Assign the first value to a variable representing the current largest number.
3. Compare the current largest with the second number using a nested if block and update when needed.
4. Within the appropriate branches, compare against the third number to ensure the final value is the maximum.
5. Print both the initial numbers and the final result for verification.

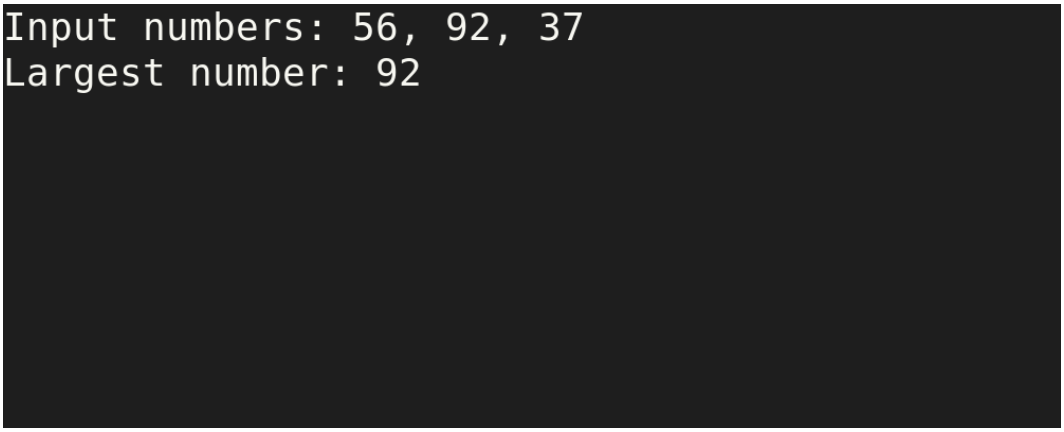
Program

```
/**
* Program: Largest of three numbers using nested if statements.
* Usage: php task1_largest.php
*/

$numbers = [56, 92, 37]; // Sample dataset; adjust as needed.

$a = $numbers[0];
$b = $numbers[1];
$c = $numbers[2];

$largest = $a;

if ($largest < $b) {
if ($b > $c) {
$largest = $b;
} else {
$largest = $c;
}
} else {
if ($largest < $c) {
$largest = $c;
```

```
}
}

echo "Input numbers: {$a}, {$b}, {$c}\n";
echo "Largest number: {$largest}\n";
```

Output

Screenshot of the CLI output captured after executing the PHP script:



Conclusion

Nested conditionals accurately identified the largest integer, demonstrating correct control flow for comparative evaluation.

Task 2: Reverse a String Using strrev()

AIM

Reverse a predefined string using PHP's built-in strrev function.

Problem Statement

Create a PHP CLI script that stores a phrase in a string variable, applies strrev() to reverse it, and prints both the original and reversed outputs.

Constraints

- The solution must use the strrev() function to handle string reversal.
- Program output needs to highlight both the original string and its reversed counterpart clearly.

Procedure

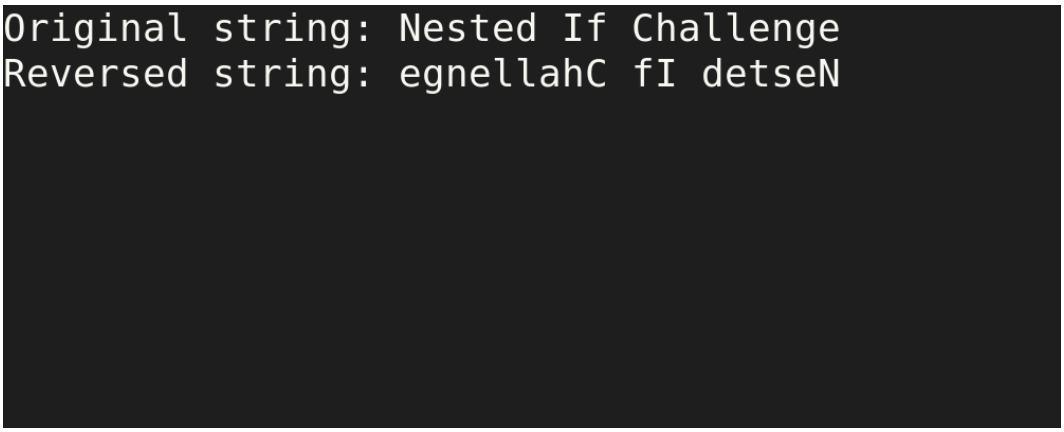
1. Declare a string variable with the phrase targeted for reversal.
2. Invoke PHP's strrev() function and store the reversed value.
3. Display the original and reversed strings on separate lines for readability.

Program

```
/**
* Program: Reverse a string using strrev.
* Usage: php task2_reverse.php
*/

$input = "Nested If Challenge";
$reversed = strrev($input);

echo "Original string: {$input}\n";
echo "Reversed string: {$reversed}\n";
```

Output

Screenshot of the CLI output captured after executing the PHP script:



Conclusion

The strrev() function reversed the input phrase precisely, confirming the utility of PHP's standard library for string manipulation.