\*\*\* Testing assignments and variable aliasing: \*\*\*

This should read "blah": \n"; ?> This should read "this is nifty": \n"; ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* Testing integer operators \*\*\*

Correct result - 8:

Correct result - 8:

Correct result - 2:

Correct result - -2:

Correct result - 15:

Correct result - 15:

Correct result - 2:

Correct result - 3:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* Testing real operators \*\*\*

Correct result - 8:

Correct result - 8:

Correct result - 2:

Correct result - -2:

Correct result - 15:

Correct result - 15:

Correct result - 2:

Correct result - 3:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* Testing if/elseif/else control \*\*\*

\n"; } elseif ($a == "5") { echo "This "." works

\n"; $a = 6; if ("andi" == ($test = "andi")) { echo "this\_still\_works

\n"; } elseif (1) { echo "should\_not\_print

\n"; } else { echo "should\_not\_print

\n"; } if (44 == 43) { echo "should\_not\_print

\n"; } else { echo "should\_print

\n"; } } elseif ($a == 6) { echo "this "."broken

\n"; if (0) { echo "this\_should\_not\_print

\n"; } else { echo "TestingDanglingElse\_This\_Should\_not\_print

\n"; } } else { echo "This "."does "." not"." work

\n"; } ?> \*\*\* Seriously nested if's test \*\*\*

\*\* spelling correction by kluzz \*\* \n"; if (0) { /\* this code is not supposed to be executed \*/ echo "hmm, this shouldn't be displayed #1

\n"; $j++; if (1) { $i += $j; if (0) { $j = ++$i; if (1) { $j \*= $i; echo "damn, this shouldn't be displayed

\n"; } else { $j /= $i; ++$j; echo "this shouldn't be displayed either

\n"; } } elseif (1) { $i++; $j++; echo "this isn't supposed to be displayed

\n"; } } elseif (0) { $i++; echo "this definitely shouldn't be displayed

\n"; } else { --$j; echo "and this too shouldn't be displayed

\n"; while ($j>0) { $j--; } } } elseif (2-2) { /\* as long as 2-2==0, this isn't supposed to be executed either \*/ $i = ++$j; echo "hmm, this shouldn't be displayed #2

\n"; if (1) { $j = ++$i; if (0) { $j = $i\*2+$j\*($i++); if (1) { $i++; echo "damn, this shouldn't be displayed

\n"; } else { $j++; echo "this shouldn't be displayed either

\n"; } } else if (1) { ++$j; echo "this isn't supposed to be displayed

\n"; } } elseif (0) { $j++; echo "this definitely shouldn't be displayed

\n"; } else { $i++; echo "and this too shouldn't be displayed

\n"; } } else { $j=$i++; /\* this should set $i to 1, but shouldn't change $j (it's assigned $i's previous values, zero) \*/ echo "this should be displayed. should be: \$i=1, \$j=0. is: \$i=$i, \$j=$j

\n"; if (1) { $j += ++$i; /\* ++$i --> $i==2, $j += 2 --> $j==2 \*/ if (0) { $j += 40; if (1) { $i += 50; echo "damn, this shouldn't be displayed

\n"; } else { $j += 20; echo "this shouldn't be displayed either

\n"; } } else if (1) { $j \*= $i; /\* $j \*= 2 --> $j == 4 \*/ echo "this is supposed to be displayed. should be: \$i=2, \$j=4. is: \$i=$i, \$j=$j

\n"; echo "3 loop iterations should follow:

\n"; while ($i<=$j) { echo $i++." $j

\n"; } } } elseif (0) { echo "this definitely shouldn't be displayed

\n"; } else { echo "and this too shouldn't be displayed

\n"; } echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\n"; } ?> \*\*\* C-style else-if's \*\*\*

\n"; } else if ($a++) { echo "This shouldn't be displayed either

\n"; } else if (--$a) { echo "No, this neither

\n"; } else if (++$a) { echo "This should be displayed

\n"; } else { echo "This shouldn't be displayed at all

\n"; } ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* WHILE tests \*\*\*

$j) { echo "$i is greater than $j

\n"; } else if ($i==$j) { echo "$i equals $j

\n"; } else { echo "$i is smaller than $j

\n"; } $i++; } ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* Nested WHILEs \*\*\*

\n"; $i=0; while ($i<$arr\_len) { $j=0; while ($j<$arr\_len) { $k=0; while ($k<$arr\_len) { echo "\${test$i$j}[$k] = ".${"test$i$j"}[$k]."

\n"; $k++; } $j++; } $i++; } ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* hash test... \*\*\*

\n"; } \*/ echo "commented out..."; ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* Hash resizing test \*\*\*

0) { $a = $a . 'a'; echo "$a

\n"; $resize[$a] = $i; $i--; } $i = 10; $a = 'b'; while ($i > 0) { $a = $a . 'a'; echo "$a

\n"; echo $resize[$a]."

\n"; $i--; } ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* break/continue test \*\*\*

\n"; while ($i<5) { if ($i>2) { break; } $j=0; echo "\$j should go from 3 to 4, and \$q should go from 3 to 4

\n"; while ($j<5) { if ($j<=2) { $j++; continue; } echo " \$j=$j

\n"; for ($q=0; $q<=10; $q++) { if ($q<3) { continue; } if ($q>4) { break; } echo " \$q=$q

\n"; } $j++; } $j=0; echo "\$j should go from 0 to 2

\n"; while ($j<5) { if ($j>2) { $k=0; echo "\$k should go from 0 to 2

\n"; while ($k<5) { if ($k>2) { break 2; } echo " \$k=$k

\n"; $k++; } } echo " \$j=$j

\n"; $j++; } echo "\$i=$i

\n"; $i++; } ?> \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* Nested file include test \*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\n"; # testing some PHP style comment... } ?>