

PHP Programming Assignment 1

1. Write a function `removeAllValuesMatching($arr, $value)` that takes an associative array and a value as a parameter and returns a new associative array in which any array value matching `$value` is removed. In other words,

```
$arr = array(  
    'a' => "one",  
    'b' => "two",  
    'c' => "three",  
    'd' => "two",  
    'e' => "four",  
    'f' => "five",  
    'g' => "three",  
    'h' => "two"  
);  
print_r(removeAllValuesMatching($arr, "two"));
```

would print:

```
Array  
(  
    [a] => one  
    [c] => three  
    [e] => four  
    [f] => five  
    [g] => three  
)
```

ANSWER:

```
function removeAllValuesMatching($arr, $value) {  
    $result = array();  
    foreach ($arr as $key => $val) {  
        if ($value !== $val) {  
            $result[$key] = $val;  
        }  
    }  
    return $result;  
}
```

2. Write a function `removeDuplicates($arr)` that takes an associative array as a parameter and returns the same associative array except with all duplicate values removed. You may not use the function `array_unique` for this problem. For example:

```
$arr = array(  
    'a' => "one",  
    'b' => "two",  
    'c' => "three",  
    'd' => "two",  
    'e' => "four",  
    'f' => "five",  
    'g' => "three",  
    'h' => "two"
```

```

'a' => "one",
'b' => "two",
'c' => "three",
'd' => "two",
'e' => "four",
'f' => "five",
'g' => "three",
'h' => "two"
);
print_r(removeDuplicates($arr));

```

Would print:

```

Array
(
    [a] => one
    [e] => four
    [f] => five
)

```

ANSWER:

```

function removeDuplicates($arr) {
    foreach ($arr as $key1 => $val1) {
        foreach($arr as $key2 => $val2) {
            if ($val1 == $val2 && $key1 != $key2) {
                unset($arr[$key2]);
            }
        }
    }
    $result = array();
    foreach ($arr as $key => $value) {
        $result[$key] = $value;
    }
    return $result;
}

```

- Write a function `mySoundex($name)` that computes the Soundex code for the given parameter name. Soundex is a set of rules that transforms an English name into a hash-code that can be used for similar names (so looking up a name in a database based on the soundex code will yield similar sounding names). The Soundex rules are given at <http://en.wikipedia.org/wiki/Soundex>. Note: PHP has a built-in `soundex()` function that you should NOT use for this assignment (the idea is to write it yourself). Test your function using the names "Euler", "Ellery", "Gauss", "Ghosh", "Hilbert", "Heilbronn", "Knuth", "Kant", "Leida", "Ladd", "Lukasiewicz", and "Lissajous"

ANSWER:

```
function mySoundex($name) {
    // zeros are skipped characters, otherwise the soundex number
    $soundmap = array(0, 1, 2, 3, 0, 1, 2, 0, 0, 2, 2, 4, 5,
        5, 0, 1, 2, 6, 2, 3, 0, 1, 0, 2, 0, 2);
    // length of the produced string
    $soundlen = 4;
    $result = '';

    // make an array, always lower case, no spaces.
    $name = str_split(strtolower(trim($name)));

    for ($i = 0, $j = 0; $j < count($name) && $i < $soundlen;
    ++$j) {
        $ch = $name[$j];
        if (isLetter($ch)) {
            // see what letter we're dealing with
            $num = $soundmap[ord($ch) - ord('a')];

            // should we use the letter?
            if ($num != 0 || $i === 0) {
                if ($i === 0) {
                    // always use 1st letter.
                    $result .= $ch;
                    ++$i;

                    // avoid repeated letters
                } else if (substr($result, $i-1, 1) != $num ) {
                    $result .= $num;
                    ++$i;
                }
            }
        }
    }

    // fill with zeros up to length
    while ($i < $soundlen) {
        $result .= '0';
        ++$i;
    }
    return $result;
}
```

- Write a function `findSpellings($word, $allWords)` that takes a string and an array of dictionary words as parameters. The function should return an array of possible spellings for a misspelled `$word`. One way to approach this is to use the `soundex()` function to find words in `$allWords` that match the soundex for `$word`.

ANSWER:

```
function findSpellings($word, $allWords) {
    $result = array();
    $match = soundex($word);
    foreach ($allWords as $possible) {
        if (strcasecmp($match, soundex($possible)) == 0) {
            $result[] = $possible;
        }
    }
    return $result;
}
```

- Write a function in PHP that receives a number (`$num`) as an input and returns true if the number is a prime number, and false if it is not prime. Use Wikipedia to research what a prime number is if you do not already know.

ANSWER:

```
function isPrime($num) {
    // primes are only positive, starting at 2
    if ($num <= 1) {
        return false;
    }

    // special case for 2 as the only even prime
    if ($num % 2 == 0 && $num != 2) {
        return false;
    }

    // largest possible factor of $num is its square root
    $stop = ceil(sqrt($num));

    // work with odd numbers for efficiency
    for ($i = 3; $i <= $stop; $i += 2) {
        if ($num % $i == 0) {
            return false;
        }
    }
}
```

```
    }  
  }  
  return true;  
}
```