# **Lab: String and Text Processing**

Problems for lab for the "PHP Fundamentals" course @ SoftUni.

You can check your solutions in Judge.

## 1. Substring

On the first line you will receive a string. On the second line you will receive a second string. Write a program that removes all of the occurrences of the first string in the second until there is no match. At the end print the remaining string.

### **Examples**

Input	Output	Comment	
ice	kgb	We remove ice once and we get "kgiciceeb"	
kicegiciceeb		We match "ice" one more time and we get "kgiceb"	
		There is one more match. The final result is "kgb"	

#### Hints

- Read the input.
- Remove the matches.
  - Use the function str\_replace(searchText, '', string)

## 2. Reverse Strings

You will be given series of strings until you receive an "end" command. Write a program that reverses strings and print word and reversed word on separate line in format "{word} = {reversed word}".

## **Examples**

Input	Output	
helLo Softuni bottle end	helLo = oLleh Softuni = inutfoS bottle = elttob	
Dog caT chAir end	Dog = goD caT = Tac chAir = riAhc	

#### **Solution**

Use while loop and read strings until you receive "end".



















```
<?php
$input = readline();
while($input != "end"){
    $input = readline();
```

Reverse the string with **strrev()** function.

```
$reversed = strrev($input);
```

Print the reversed string in the specified format.

```
echo $input . " = " . $reversed . PHP_EOL;
```

## 3. Repeat strings

Write a program that reads an array of strings. Each string is repeated **n** times, where **n** is the length of the string. Print the concatenated string.

### **Examples**

Input	Output	
hi abc add	hihiabcabcabcaddaddadd	
work	workworkwork	
ball	ballballball	

#### Solution

Read a string array.

```
$words = explode( delimiter: " ", readline());
```

Iterate through elements in the array.

```
foreach ($words as &$word) {
```

Find the length of the current word.

```
foreach ($words as &$word) {
    $count = strlen($word);
    echo str repeat ($word, $count);
```















#### 4. Text Filter

Write a program that takes a text and a string of banned words. All words included in the ban list should be replaced with asterisks "\*", equal to the word's length. The entries in the ban list will be separated by a comma and space ", ".

The ban list should be entered on the first input line and the text on the second input line.

### **Examples**

Input	Output
functionality Therefore we owe it to them by	It is not *****, it is GNU/****. **** is merely the kernel, while GNU adds the functionality. Therefore we owe it to them by calling the OS GNU/****! Sincerely, a ****** client

#### Hints

- Read the input.
- Replace all ban words in the text with asterisk (\*).
  - Use the function str replace().
  - Use str\_repeat() to create the replacement

## 5. Count string occurrences

Write a program that receives a text and a string to search for. Use spaces, commas, dots, question marks and exclamation marks as word delimiters. Print all the occurrences of that word in the string

## **Examples**

Input	Output
This is a word and it also is a sentence. is	2
How are you?? Good, thanks.	1

#### Hint

Split the sentence into words and create a counter that stores how many times the searched word occurs

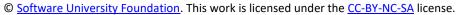
### 6. Palindromes

Write a program that extracts from a given text all palindromes, e.g. ABBA, lamal, exe and prints them on the console on a single line, separated by comma and space. Use spaces, commas, dots, question marks and exclamation marks as word delimiters. Print only unique palindromes, sorted natural order.

## **Examples**

Input	Output	
Hi,exe? ABBA! Hog fully a string. Bob	a, ABBA, exe	



















### Hints

To sorting result use **natcasesort()** function

















