Цель домашнего задания

Научиться решать алгоритмические задачи на php.

Что входит в задание

1. Алгоритмическая задача на подсчет количества букв в слове.
2. Алгоритмическую задача на поиск минимального элемента в массиве.
3. Алгоритмическую задача на подсчет суммы элементов массива.
4. Алгоритмическую задача на определение местоположения машин.
5. Частная практическая задача на склонение слова-окончания в зависимости от числа.

Задание 1

Что нужно сделать

Реализуйте подсчёт различных букв в слове. Создайте переменную $line, поместите в неё любой текст. Напишите алгоритм, который подсчитает, сколько в этой переменной содержится различных символов (большие и маленькие буквы считайте за разные символы).

Например, для строки Student, hello! должен быть сформирован такой результат:

|  |
| --- |
| S - 1 t - 2 u - 1 d - 1 e - 2 n - 1 , - 1  - 1 h - 1 l - 2 o - 1 ! - 1 |

Советы и рекомендации

* Алгоритм решается напрямую, в лоб. Пройдите циклом по всем символам и постепенно прибавляйте количество нужных символов.

Что оценивается

* Алгоритм должен решать требуемую задачу.
* Алгоритм не должен быть завязан именно на ту строку, которую вы укажете в переменной $line.

Задание 2

Что нужно сделать

Реализуйте алгоритм поиска минимального элемента в массиве.

1. Создайте массив $values, содержащий десять случайных чисел от 0 до 100.
2. Найдите в этом массиве минимальное значение.
3. Выведите сам массив, минимальное значение и его позицию в массиве. Если таких значений несколько, то выведите первое из них.

Советы и рекомендации

* Алгоритм поиска минимума точно такой же, что и алгоритм поиска максимума.

Что оценивается

* Алгоритм должен решать требуемую задачу.
* Для решения задачи не должна использоваться встроенная функция min.

Задание 3

Что нужно сделать

Реализуйте алгоритм подсчёта суммы элементов массива.

1. Создайте массив $numbers случайной длины от 3 до 20. Каждое значение — это случайное число от 0 до 10.
2. Посчитайте сумму его элементов, стоящих под нечётным индексом.
3. Выведите этот массив и полученную сумму.

Советы и рекомендации

* Можно пройтись циклом по всем элементам, пропуская те, что стоят на нечётных местах, а можно составить цикл так, чтобы он проходил только по нечётным позициям.

Что оценивается

* Алгоритм должен решать требуемую задачу.
* Для решения задачи не должна использоваться встроенная функция min.

Задание 4

Что нужно сделать

**Исходные данные**

В неизвестном мире существует шоссе. Это шоссе растянулось от 0 до 1000 км (а может, и дальше, мы не знаем, так далеко ещё никто не ходил). На нём в случайных местах находятся два города. На шоссе действует строгое ограничение скорости: 70 км/ч в городе и 90 км/ч за городом.

**Дано:**

* **$city1** — километр шоссе, на котором расположен центр первого города;
* **$city1Radius** — радиус первого города, задается в км (наши города — это идеальный круг);
* **$city2**— километр шоссе, на котором расположен центр второго города;
* **$city2Radius** — радиус второго города, задается в км.

Создайте перечисленные переменные и положите в них целые значения: придумайте их сами или генерируйте случайные числа.

**Задача**

По шоссе едут десять машин, для каждой известен километр шоссе (случайное целое число от 0 до 1000), на котором машина находится в данный момент. Все автолюбители строго соблюдают скоростной режим и всегда едут с максимальной разрешённой скоростью. Для каждой машины выведите подобную строку: *"Машина 4 едет по городу на 3 км со скоростью не более 70"*.

Советы и рекомендации

* Нарисуйте шашлык на шампуре — он отлично проиллюстрирует условие задания.
* Не пытайтесь рассматривать, где города находятся друг относительно друга. Для самого простого решения этой задачи их расположение не имеет значения. Пусть даже один город расположен полностью в другом.
* Машин десять, но нужно создавать десять разных переменных. Для выполнения одинаковых действий следует использовать циклы. Представьте, что их число можно поменять. От этого ваш код не должен измениться, он должен работать для любого количества машин, за счет простого изменения одной цифры.

Что оценивается

* Стили оформления кода должны быть соблюдены.
* Код не должен зависеть от порядка расположения городов.
* Если по шоссе нужно запустить ещё одну машину, код не придется дорабатывать.

Задание 5

Что нужно сделать

Классическое задание на склонение окончания слова в зависимости от числа.

1. Создайте переменную $studentsCount и присвойте ей случайное значение от 1 до 1000000.
2. Создайте программу, которая выведет в нужной форме текстовое сообщение. Например: "На учёбе 100 студентов", "На учебе 2 студента" и т. д.

Советы и рекомендации

* Найдите закономерность, представьте алгоритм и переведите его в код.
* Помните: почти из любого правила есть исключения. В этой задаче без них тоже не обошлось.
* Можно легко найти готовое решение этой задачи, но разве так интересно? :)

Что оценивается

* Стили оформления кода должны быть соблюдены.
* Слово «студент» должно выводиться в корректной форме для любого числа из указанного диапазона.
* Код не должен дублироваться. В нём не должно быть одинаковых или взаимоисключающих условий.

Критерии оценки

* Каждое задание запускается и работает корректно. Ошибки при их запуске не появляются.
* Соблюдены стили оформления кода (psr-1 и psr-12).
* Выполнены все шесть заданий.