Первое задание подразумевает написание функции, реализующей описанный функционал. По возможности – входные данные генерируем программно, случайным образом.

Во втором задании – обязательно применение регулярных выражений.

***ЗАДАНИЕ 1***

***Вариант 1.*** Текст произвольного содержания задан в виде последовательности (списка) слов. Написать функцию, определяющую словарный запас текста, т.е. число уникальных слов.

***Вариант 2.*** Текст произвольного содержания задан в виде последовательности (списка) слов. Написать функцию, определяющую частоту вхождения отдельных слов в этот текст.

***Вариант 3.*** Напишите функцию проверки целого числа на принадлежность его к множеству простых чисел и функцию нахождения списка простых чисел меньших или равных заданному.

***Вариант 4.*** Напишите функцию сортировки списка вещественных чисел методом “пузырька”

***Вариант 5.*** Напишите функцию, реализующую операцию проверки на включение над множествами (множества заданы в виде списков): множество A включает множество B, если в нем присутствуют все элементы из множества B.

***Вариант 6.*** Напишите функцию, реализующую операцию пересечения множеств: C = A ∩ B. Пересечение множеств содержит все элементы, присутствующие одновременно в обоих множествах. Например “1,2,3,4,6,7,8” ∩ “2,4,5,8,9” равно “2,4,8”.

***Вариант 7.*** Напишите функцию, реализующую операцию объединения множеств: C = A ∪ B. Объединение множеств содержит все элементы, присутствующие в обоих множествах без повторов. Например “1,2,3,4,6,7,8” ∪ “2,4,5,8,9” равно “1,2,3,4,5,6,7,8,9”.

***Вариант 8.*** Напишите функции, реализующие следующие операции над векторами:

* Определение скалярного произведения двух векторов
* Приведение вектора к единичной длине

***Вариант 9.*** Напишите функцию, реализующую операцию матричного умножения.

***Вариант 10.*** Напишите функцию, реализующую операцию умножения вектора на матрицу

***Вариант 11.*** Напишите функцию, вычисляющую значение полинома y(x) = a0+ a1 x+a2 x2+a3 x3+…+an xn от заданного x. Полином задается списком коэффициентов [a0,a1,...,an].

***Вариант 12.*** Напишите функцию, вычисляющую дисперсию δ выборки X, заданной списком вещественных чисел [x1,x2,...,xn]: δ = ((x1 − xM)2 + (x2 − xM)2 + ... + (xn − xM)2 ) / (n – 1), xM возьмем равным среднему арифметическому от выборки.

***Вариант 13.*** Дан список строк и словарь, содержащий пары “слово”: “замена”. Разработайте функцию, последовательно применяющую все замены из словаря к списку, и возвращающую новый список.

***Вариант 14.*** Дан список словарей, имитирующий базу данных, в которой каждый словарь считается записью. Разработайте функцию, осуществляющую поиск в базе данных всех записей, в которых определенное поле принимает заданное значение. Поле и его значение задаются в параметрах функции.

***Вариант 15.*** Разработайте функцию, получающую два словаря и возвращающую список, содержащий значения из первого словаря, являющихся ключами во втором и значения из второго, являющихся ключами в первом.

***Вариант 16.*** Даны четыре действительных числа: *x*1, *y*1, *x*2, *y*2. Напишите функцию distance(x1, y1, x2, y2), вычисляющую расстояние между точкой (*x*1http://informatics.mccme.ru/moodle/lib/jsMath/fonts/cmmi10/alpha/120/char3B.png*y*1) и (*x*2http://informatics.mccme.ru/moodle/lib/jsMath/fonts/cmmi10/alpha/120/char3B.png*y*2). Считайте четыре действительных числа и выведите результат работы этой функции.

***Вариант 17.*** По данному числу N выведите все строки длины N из нулей и единиц в лексикографическом порядке.

***ЗАДАНИЕ 2***

***Вариант 1.*** Разработайте функцию, переформатирующую входной параметр-текст по следующим правилам:

* слева от знаков препинания не должно быть пробелов;
* между словами не должно быть более одного пробела;
* после завершающей предложение точки ставятся два пробела.

***Вариант 2.*** Разработайте функцию, переформатирующую текст, заданный в ее первом параметре так, чтобы каждая строка была не длиннее, чем n символов. Число n задается во втором параметре. Отдельные слова при этом не должны разрываться. Текст может быть разбит на абзацы, форматируемые независимо друг от друга. Разделитель абзацев — пустая строка.

***Вариант 3.*** Разработайте функцию, объединяющую абзацы во входном тексте в одну строку. Абзацы могут задаваться отступом в начале строки или пустой строкой. В результирующих строках-абзацах не должно быть лишних пробелов между словами.

***Вариант 4.*** Разработайте функцию, переформатирующую непрерывный (без абзацев) текст, заданный в ее первом параметре так, чтобы каждая строка была не длиннее, чем n символов. При необходимости, часть слова может быть перенесена на следующую строку, при этом разрыв можно осуществлять по слогам, перечисленным в отдельном списке- параметре.

***Вариант 5.*** Разработайте функцию, форматирующую текст, заданный в виде длинных строк, разделенных символом перевода строки, в абзацы заданной ширины, не разбивая отдельные слова. Каждая исходная строка образует отдельный абзац, первая строка которого имеет отступ в 4 пробела.

***Вариант 6.*** Разработайте функцию, форматирующую текст на входе по правилам:

* операторы ’+’, ’-’, ’\*’, ’/’ отделяются от других слов пробелами;
* круглые скобки не должны иметь пробелов после открывающей скобки и перед закрывающей.

***Вариант 7.*** В языке разметки HTML есть понятие тэг и атрибут. В строке:

<a href="http://www.quackit.com/html/tutorial/html\_links.cfm">Example Link</a>

a это имя тега, href это имя атрибута, <http://www.quackit.com/html/tutorial/html_links.cfm> - значение атрибута. Составьте регулярное выражение, принимающее на вход строку с html-рамзеткой (набором тегов), а на выходе генерирующее список имён тегов, список имён атрибутов, список значений атрибутов

***Вариант 8.*** Составьте регулярное выражение, проверяющее на валидность номер телефона в формате +7 (910) 910-91-091

***Вариант 9.*** Составьте регулярное выражение, проверяющее на валидность IPv4 адрес. Формат IPv4 адреса - A.B.C.D где A, B, C и D целые числа между 0 и 255 (включительно).

***Вариант 10.*** Составьте регулярное выражение, проверяющее на валидность IPv6 адрес. Формат IPv6 адреса – 8 групп, разделённых знаком «:», в каждой из которых по 4 16-ричных знака. Например, 2001:0db8:0000:0000:0000:ff00:0042:8329. Лидирующие нули каждой группы могут быть пропущены, т.е. предыдущий пример может быть записан как 2001:db8:0:0:0:ff00:42:8329

***Вариант 11.*** Составить регулярное выражение, вычленяющее из текста все валидные email-адреса.

***Вариант 12.*** Составить регулярное выражение, вычленяющее из текста все валидные доменные имена 2 и 3 уровня в зонах ru, org, com, net.

***Вариант 13.*** Слова в некотором выдуманном языке могут состоять только из букв **a**, **b**, **c**, и при этом:

* никогда не содержат двух букв **b** подряд,
* ни в одном слове никогда не встречается три одинаковых подслова подряд. Например, по этому правилу в наш язык не могут входить слова **aaa** (так как три раза подряд содержит подслово **a**), **ababab** (так как три раза подряд содержит подслово **ab**), **aabcabcabca** (три раза подряд содержит подслово **abc**).

Все слова, удовлетворяющие вышеописанным правилам, входят в наш язык.

Напишите функцию, которая по данному слову определит, принадлежит ли оно этому языку.

***Вариант 14.*** Напишите функцию, которая посчитает количество смайликов в заданном тексте.

Смайликом будем считать последовательность символов, удовлетворяющую условиям:

\* первым символом является либо ; (точка с запятой) либо : (двоеточие) ровно один раз

\* далее может идти символ – (минус) сколько угодно раз (в том числе символ минус может идти ноль раз)

\* в конце обязательно идет некоторое количество (не меньше одной) одинаковых скобок из следующего набора: (, ), [, ].

\* внутри смайлика не может встречаться никаких других символов.

Например, нижеприведенные последовательности являются смайликами:

:)

;---------[[[[[[[[

в то время как эти последовательности смайликами не являются (хотя некоторые из них содержат смайлики):

:-)]

;--

-)

::-(

:-()