**ВАРИАНТЫ ИНДИВИДУАЛЬНЫХ ЗАДАНИЙ**

**Задание 1**. Составьте шаблон регулярного выражения для проверки корректного ввода номера паспорта гражданина Республики Беларусь и его личного идентификационного номера (разными вариантами регулярного выражения). Протестируйте шаблон.

**Задание 2.** Выполнить индивидуальное задание согласно варианту, выданному преподавателем

Вариант 1

1. Отыскать любое слово, расположенное между парой тегов <b> и </b> HTML, без включения этих тегов в общее соответствие регулярному выражению. Например, для испытуемого текста My <b>cat</b> is furry, правильным соответствием будет cat.
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. В строке с телефонными переговорами клиента определить по какому номеру выполнены подряд (2 и более соединений) звонки.
3. В текстовом файле могут содержаться (как отдельные слова) MAC-адреса. Вывести все такие слова, третье число в которых равно искомому, в новый файл (по одному в строку). Искомое число вводится с клавиатуры в десятичной системе счисления. MAC-адрес состоит из шести двузначных 16-ричных чисел, разделённых двоеточием. Регистр букв не различается. Пример правильного MAC-адреса: aE:dC:cA:56:76:54

Вариант 2

1. Создать регулярное выражение, совпадающее со списком слов one, two и three, разделенных запятыми. Каждое слово может присутствовать в списке не менее одного раза.
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. Вывести на экран монитора строку, содержащую максимальное количество повторяющихся слов.
3. Найти количество строк текстового файла, которые заканчиваются словом, являющимся правильным MAC-адресом. MAC-адрес состоит из шести двузначных 16-ричных чисел, разделённых двоеточием. Регистр букв не различается. Пример правильного MAC-адреса: aE:dC:cA:56:76:54

Вариант 3

1. Используя регулярные выражения, написать программу, которая следующую задачу. Время имеет формат часы:минуты. И часы, и минуты состоят из двух цифр, пример: 09:00. Напишите регулярное выражение для поиска времени в строке: «Завтрак в 09:00». Учтите, что «37:98» – некорректное время.
2. Задан текст, в котором находятся номера телефонов в формате ххх-хх-хх. Используя регулярные выражения, найти и вывести эти номера.
3. В текстовом файле могут содержаться (как отдельные слова) числа в римской системе счисления. Вывести значения всех таких слов в десятичной системе счисления в порядке возрастания. Если таких слов нет, вывести соответствующее сообщение.

Вариант 4

1. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь телефон в формате 8-(код)номер. Например, 8-(29)9996699. Учтите, что 32 и т.д. не правильный мобильный код для нашей страны.
2. Текст состоит из слов, чисел и знаков препинания. Составить регулярное выражение для поискав нем слов. Вывести их на экран консоли.
3. «Хороший» пароль должен иметь длину в 8 символов, содержать большие, маленькие буквы латинского алфавита, цифры, подчёркивание, причём должен быть включён хотя бы один символ из каждой группы. Найти количество «хороших» паролей в текстовом файле.

Вариант 5

1. Выполнить поиск с заменой, в процессе которого все адреса URL будут преобразованы в ссылки HTML, указывающие на эти адреса, и использовать обнаруженные адреса URL как замещающий текст. Для данного упражнения примем, что адреса URL начинаются с последовательности ≪http:≫, за которой следуют любые непробельные символы. Например, текст Please visit http://www.regexcookbook.com должен превратиться в текст Please visit <a href=”http://www.regexcookbook.com”>http://www.regexcookbook.com</a>.
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. В строке с телефонными переговорами клиента определить все номера, время связи с которыми было менее 5 секунд.
3. Некоторые строки текстового файла содержат GUID (строчка, состоящая из 8, 4, 4, 4, 12 шестнадцатеричных цифр, разделенных символом «минус»). Регистр букв не различается. Создать на основе входного файла новый текстовый файл, не включая в него найденные строки. Пример GUID: e02fd0e4-00fd-090A-ca30-0d00a0038ba0

Вариант 6

1. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь HTML-цвет. Цвет задается в формате #ABCDEF, то есть # и содержит затем 6 шестнадцатеричных символов.
2. Задан текст, который состоит из слов, чисел и знаков препинания. Заменить каждое вхождение слова «кот» на слово «котик». Вывести результат на экран консоли.
3. Найти количество строк текстового файла, которые содержат не менее двух слов, являющихся правильной записью времени в формате HH:MM:SS

Вариант 7

1. Используя регулярные выражения, написать программу, которая решает следующую задачу. Арифметическое выражение состоит из двух чисел и операции между ними, например: 1 + 2, 1.2 \*3.4, -3/ -6, -2-2. Список операций: «+», «-», «\*» и «/». Также могут присутствовать пробелы вокруг оператора и чисел. Напишите регулярное выражение, которое найдёт как всё арифметическое действие, так и два числа-участника операции.
2. Задан текст, который состоит из слов, чисел и знаков препинания. Заменить каждое вхождение слова «лаба» на пробел. Вывести результат на экран консоли.
3. В текстовом файле могут содержаться (как отдельные слова) числа в римской системе счисления. Вывести значение максимального из таких слов в десятичной системе счисления. Если таких слов нет, вывести ноль.

Вариант 8

1. Используя регулярные выражения, написать программу, которая решает задачу проверки имени заливаемого графического объекта на хостинг.
2. Используя регулярные выражения, написать программу, которая удаляет из строки все, что находится после символа #. Вывести результат на экран консоли.
3. Найти количество строк текстового файла, которые не содержат слов, являющихся правильной записью времени в 12-часовом формате (часы и минуты разделяются двоеточием, секунды отсутствуют).

Вариант 9

1. Создать регулярное выражение, которое определяет вещественное число с необязательной целой частью, обязательной дробной частью и необязательной экспонентой. В каждой части допускается любое число цифр.
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. В строке с телефонными переговорами клиента выполнить сортировку переговоров по их стоимости.
3. Формат даты dd MMM yy содержит: номер дня без ведущего нуля, пробел, трёхсимвольное английское обозначение месяца (прописными буквами), пробел, две последние цифры года. В текстовом файле могут содержаться (как отдельные слова) записи дат в таком формате. Найти максимальную дату, предполагая, что все они относятся к XX столетию. Если таких слов нет, вывести соответствующее сообщение.

Вариант 10

1. Используя регулярные выражения, написать программу, которая решает задачу проверки верно ли пользователь ввел адрес id сайта Вконтакте (в старом формате, например, id78195751).
2. Найти в тексте, состоящем из слов, цифр и знаков препинания, все слова, длина которых больше 7 символов. Вывести результат на экран консоли.
3. Найти количество строк текстового файла, которые начинаются словом, являющимся правильной датой в формате dd/mm/yyyy. Диапазон лет - от 1800 до 2100. Правильно обрабатывать високосные года.

Вариант 11

* 1. Создать регулярное выражение, которому соответствовали бы любые ошибочные написания слова calendar, чтобы иметь возможность отыскивать это слово в документе, не полагаясь на грамотность автора. Предполагается, что на месте любой гласной буквы может использоваться символ a или e. Создать второе регулярное выражение, которому соответствовала бы единственная шестнадцатеричная цифра. Создать третье регулярное выражение, которому соответствовал бы единственный символ, не являющийся шестнадцатеричной цифрой.
  2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. В строке с телефонными переговорами клиента определить по какому номеру выполнено максимальное количество звонков.
  3. «Хороший» пароль должен иметь длину в 8 символов, содержать большие, маленькие буквы латинского алфавита, цифры, подчёркивание, причём должен быть включён хотя бы один символ из каждой группы. Найти количество «хороших» паролей в текстовом файле.

Вариант 12

1. Создать регулярное выражение, совпадающее с ≪магическими≫ датами в формате yyyy-mm-dd. Магической считается дата, когда последние две цифры года, месяц и день являются одним и тем же числом. Например, магической считается дата 2008-08-08. Будем исходить из предположения, что все даты, присутствующие в испытуемом тексте, корректны. Регулярное выражение не должно исключать такие даты, как 9999-99-99, так как они вообще не будут появляться в испытуемом тексте. Требуется отыскать только магические даты.
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. Задана строка. Вывести на экран монитора строку, содержащую максимальное количество знаков пунктуации.
3. В текстовом файле могут содержаться (как отдельные слова) IP-адреса компьютеров в формате d.d.d.d, где d - целое число из диапазона от 0 до 255. Создать новый файл, удаляя IP-адреса, в которых последнее число d начинается с заданной цифры (данная цифра вводится с клавиатуры).

Вариант 13

1. Используя регулярные выражения, написать программу, которая решает задачу проверки верно ли пользователь ввел адрес сайта.
2. Задан текст, в котором записаны ФИО в формате (Иванов П.П.). Используя регулярные выражения, вывести все ФИО в таком формате.
3. В текстовом файле некоторые слова являются IP-адресами компьютеров в формате d.d.d.d, где d - целое число из диапазона от 0 до 255. Вывести все такие слова (по одному в строку) в новый файл.

Вариант 14

1. Создать регулярное выражение, которое определяет 32-битное шестнадцатеричное число с необязательным суффиксом h.
2. Считать текст из файла и вывести его на экран монитора, заменив цифры словами, например, «0» на слово «ноль»; «1» на слово «один» и т.д.
3. В текстовом файле могут содержаться (как отдельные слова) числа в римской системе счисления. Вывести значение максимального из таких слов в десятичной системе счисления. Если таких слов нет, вывести ноль.

Вариант 15

1. Создать регулярное выражение, совпадающее с парой тегов <p> и </p> языка разметки XHTML и текстом между ними. Текст между этими тегами может содержать другие теги XHTML.
2. Задана строка. Вывести на экран монитора строку, содержащую максимальное количество чисел (не цифр).
3. Найти количество строк текстового файла, которые содержат не менее двух слов, являющихся правильной записью времени в формате HH:MM:SS

Вариант 16

1. Создать единственное регулярное выражение, совпадающее с целым файлом HTML, которое будет проверять наличие тегов html, head, title и body и их вложенность. Выражение не должно совпадать с файлами HTML, в которых отсутствуют требуемые теги.
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. В строке с телефонными переговорами клиента определить по какому номеру был самый продолжительный по времени разговор.
3. Найти количество строк текстового файла, которые не содержат слов, являющихся правильной записью времени в 12-часовом формате (часы и минуты разделяются двоеточием, секунды отсутствуют).

Вариант 17

1. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь телефон в формате +375(код)номер. Например, +375(44)9996699. Учтите, что 32 и т.д. не правильный мобильный код для нашей страны.
2. Составить регулярное выражение для поиска в тексте повторяющихся слов, расположенных подряд и разделенных произвольным количеством пробелов, независимо от регистра. Вывести их на экран консоли.
3. Автомобильный номер Германии состоит из кода региона (от одной до трёх прописных латинских букв), пробела, кода серии (одна или две прописных латинских буквы) и собственно номера (четырёхзначного числа без ведущих нулей). Серии SS и SD не используются, как напоминающие о нацистском прошлом. Текстовый файл содержит список автомобильных номеров, по одному в строке. Найти количество ошибочных номеров.

Вариант 18

1. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь e-mail адрес в соответствующем формате.
2. Задан текст с некоторыми словами, заключенными в фигурные скобки. Используя регулярные выражения, создать программу для получения всех подстрок между круглыми скобками.
3. В текстовом файле могут содержаться (как отдельные слова) MAC-адреса. Вывести все такие слова, третье число в которых равно искомому, в новый файл (по одному в строку). Искомое число вводится с клавиатуры в десятичной системе счисления. MAC-адрес состоит из шести двузначных 16-ричных чисел, разделённых двоеточием. Регистр букв не различается. Пример правильного MAC-адреса: aE:dC:cA:56:76:54

Вариант 19

1. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь IP-адрес. Адрес состоит из четырех групп цифр, разделенных точками. Каждая группа может включать от одной до трех цифр. Примеры: 212.46.197.69, 212.194.5.106.
2. Текст состоит из слов, чисел и знаков препинания. Составить регулярное выражение для поискав нем чисел. Вывести их на экран консоли.
3. Найти количество строк текстового файла, которые начинаются словом, являющимся правильной датой в формате dd/mm/yyyy. Диапазон лет - от 1800 до 2100. Правильно обрабатывать високосные года.

Вариант 20

1. Используя регулярные выражения, написать программу, которая следующую задачу. Время имеет формат часы:минуты. И часы, и минуты состоят из двух цифр, пример: 09:00. Напишите регулярное выражение для поиска времени в строке: «Завтрак в 09:00». Учтите, что «37:98» – некорректное время.
2. Задана строка. Вывести на экран монитора только цитаты текста этой строки, то есть предложения, заключенные в кавычки.
3. В текстовом файле могут содержаться (как отдельные слова) IP-адреса компьютеров в формате d.d.d.d, где d - целое число из диапазона от 0 до 255. Создать новый файл, удаляя IP-адреса, в которых последнее число d начинается с заданной цифры (данная цифра вводится с клавиатуры).

Вариант 21

1. Задана строка. Вывести ее на экран монитора только предложения, содержащие информацию о 1000 и более «у.е.».
2. Задана строка, в которой содержаться: номер абонента, с которым выполнялось соединение; дата и время соединения; продолжительность соединения; стоимость переговоров. В строке с телефонными переговорами клиента определить по какому номеру был самый продолжительный по времени разговор.
3. Формат даты dd MMM yy содержит: номер дня без ведущего нуля, пробел, трёхсимвольное английское обозначение месяца (прописными буквами), пробел, две последние цифры года. В текстовом файле могут содержаться (как отдельные слова) записи дат в таком формате. Найти максимальную дату, предполагая, что все они относятся к XX столетию. Если таких слов нет, вывести соответствующее сообщение.

Вариант 22

1. Создать регулярное выражение, совпадающее с парой тегов <p> и </p> языка разметки XHTML и текстом между ними. Текст между этими тегами может содержать другие теги XHTML.
2. Создать регулярное выражение, совпадающее со списком слов one, two и three, разделенных запятыми. Каждое слово может присутствовать в списке не менее одного раза.
3. Найти количество строк текстового файла, которые начинаются словом, являющимся правильной датой в формате dd/mm/yyyy. Диапазон лет - от 1800 до 2100. Правильно обрабатывать високосные года.

Вариант 23

1. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь HTML-цвет. Цвет задается в формате #ABCDEF, то есть # и содержит затем 6 шестнадцатеричных символов.
2. Используя регулярные выражения, написать программу, которая проверит ввел ли пользователь телефон в формате 8-(код)номер. Например, 8-(29)9996699. Учтите, что 32 и т.д. не правильный мобильный код для нашей страны.
3. Найти количество строк текстового файла, которые содержат не менее двух слов, являющихся правильной записью времени в формате HH:MM:SS

Вариант 24

1. Создать регулярное выражение, которое определяет вещественное число с необязательной целой частью, обязательной дробной частью и необязательной экспонентой. В каждой части допускается любое число цифр.
2. Выполнить поиск с заменой, в процессе которого все адреса URL будут преобразованы в ссылки HTML, указывающие на эти адреса, и использовать обнаруженные адреса URL как замещающий текст. Для данного упражнения примем, что адреса URL начинаются с последовательности ≪http:≫, за которой следуют любые непробельные символы. Например, текст Please visit http://www.regexcookbook.com должен превратиться в текст Please visit <a href=”http://www.regexcookbook.com”>http://www.regexcookbook.com</a>.
3. В текстовом файле могут содержаться (как отдельные слова) числа в римской системе счисления. Вывести значения всех таких слов в десятичной системе счисления в порядке возрастания. Если таких слов нет, вывести соответствующее сообщение.

Вариант 25

1. Создать регулярное выражение, которое определяет 32-битное шестнадцатеричное число с необязательным суффиксом h.
2. Задана строка. Вывести на экран монитора только цитаты текста этой строки, то есть предложения, заключенные в кавычки.
3. Некоторые строки текстового файла содержат GUID (строчка, состоящая из 8, 4, 4, 4, 12 шестнадцатеричных цифр, разделенных символом «минус»). Регистр букв не различается. Создать на основе входного файла новый текстовый файл, не включая в него найденные строки. Пример GUID: e02fd0e4-00fd-090A-ca30-0d00a0038ba0

Вариант 26

1. Задана строка. Вывести ее на экран монитора только предложения, содержащие информацию о 1000 и более «у.е.».
2. Используя регулярные выражения, написать программу, которая удаляет из строки все, что находится после символа #. Вывести результат на экран консоли.
3. Автомобильный номер Германии состоит из кода региона (от одной до трёх прописных латинских букв), пробела, кода серии (одна или две прописных латинских буквы) и собственно номера (четырёхзначного числа без ведущих нулей). Серии SS и SD не используются, как напоминающие о нацистском прошлом. Текстовый файл содержит список автомобильных номеров, по одному в строке. Найти количество ошибочных номеров.

Вариант 27

1. Задана строка. Вывести ее на экран монитора только предложения, содержащие информацию о 1000 и более «у.е.».
2. Задан текст с некоторыми словами, заключенными в фигурные скобки. Используя регулярные выражения, создать программу для получения всех подстрок между круглыми скобками.
3. В текстовом файле некоторые слова являются IP-адресами компьютеров в формате d.d.d.d, где d - целое число из диапазона от 0 до 255. Вывести все такие слова (по одному в строку) в новый файл.

Вариант 28

1. Используя регулярные выражения, написать программу, которая решает задачу проверки имени заливаемого графического объекта на хостинг.
2. Задана строка. Вывести на экран монитора только цитаты текста этой строки, то есть предложения, заключенные в кавычки.
3. Найти количество строк текстового файла, которые заканчиваются словом, являющимся правильным MAC-адресом. MAC-адрес состоит из шести двузначных 16-ричных чисел, разделённых двоеточием. Регистр букв не различается. Пример правильного MAC-адреса: aE:dC:cA:56:76:54