**Отчет по лабораторной работе №1**

**Обработка текстовых данных**

## Регулярные выражения

**Задание 1 (0,5 балла)**

Найти в тексте запись об указании времени в формате чч:мм:сс. В заданном формате чч - целое число из диапазона от 00 до 23, мм и сс - целые числа из диапазона от 00 до 59 (если какая-то часть формата нарушена, то данная подстрока в качестве верного значения не рассматривается).

Регулярное выражение: (?:[0-1]\d|[2][0-3])(?::[0-5]\d){2}

Решение: <https://regex101.com/r/II0Qk2/1>

**Задание 2 (0,5 балла)**

Найти в тексте знаки препинания (точка, запятая, точка с запятой, двоеточие, тире («минус», длинное и короткое), вопросительный и восклицательный знаки, многоточие (три идущие подряд точки и отдельный символ «…» ) .

Регулярное выражение: [.,;:\-–!?…]

Решение: <https://regex101.com/r/umeM7K/1>

**Задание 3 (0,5 балла)**

В тексте могут содержаться IP-адреса компьютеров в формате d.d.d.d, где d - целое число из диапазона от 0 до 255. Найти все IP-адреса, содержащиеся в тексте.

Регулярное выражение: (?<= |^)(?:(?:\d|[1-9]\d|[1]\d\d|[2](?:[0-4]\d|[5][0-5]))\.){3}(?:(?:\d|[1-9]\d|[1]\d\d|[2](?:[0-4]\d|[5][0-5])))(?!\.?[\d\w|])

Решение: <https://regex101.com/r/SZIWA4/1>

**Задание 4 (0,5 балла)**

Проверить надежно ли составлен пароль. Пароль считается надежным, если он состоит из 8 или более символов. Пароль должен содержать хотя бы одну заглавную букву, одну маленькую букву одну цифру, хотя бы один специальный символ («!», «@», «#», «$», «%», «^», «&», «\*») и не является последовательностью («123456Qwerty», «Ps123456789», «Qwerty1»).

Регулярное выражение: (?=.\*[A-Z])(?=.\*[a-z])(?=.\*\d)(?=.\*[!@#$%^&\*])[A-z\d!@#$%^&\*]{8,}

Решение: <https://regex101.com/r/eofesN/1>

**Задание 5 (1 балл)**

Выделить в тексте прямую речь с опорой на пунктуацию (см.<http://www.gramota.tv/biblio/textbooks/gramota/punct/343-45-192?ysclid=lne281u9yq246756449>).

Регулярное выражение: (?:(?:[А-Я].\*?: )?[«\"][А-Я](?:.\*?»\.|(?:.\*?»,? – .\*?[.?!])|.\*?[!?][»\"])|(?:– [А-Я].\*?[.?!]))

Решение: <https://regex101.com/r/siGqxp/1>

**Задание 6 (1 балл)**

Выделить в тексте предложения с обобщающими словами. Рассмотреть все варианты пунктуации. (Например: «Всякий день стал приносить старый Мосеич разную крупную рыбу: щук, язей, голавлей, линей, окуней.») (см.<http://gramota.ru/class/coach/punct/45_190?ysclid=lne2kwryum489092268>).

Регулярное выражение: \b[^.?!\n]\*?: (?:[\w\- ]+, )+[\w ]+(?: – .\*?)?[!?.]|\b(?:[\w\- ]+, )+[\w ]+ – .\*?[!?.]

Решение: <https://regex101.com/r/Xecnjk/1>

**Задание 7 (1 балл)**

Дан текст с географическими названиями населенных пунктов (г. Пермь, п. Скобелевка, …). Найти вхождения населенных пунктов:

- г - город;

- пгт - поселок городского типа;

- рп - рабочий поселок;

- кп - курортный поселок;

- к - кишлак;

- дп - дачный поселок;

- п - поселок;

- п. ст - поселок при станции (поселок станции);

- с - село;

- д - деревня;

- сл - слобода;

- ст - станция;

- ст-ца - станица;

- х - хутор;

- у - улус;

- рзд - разъезд;

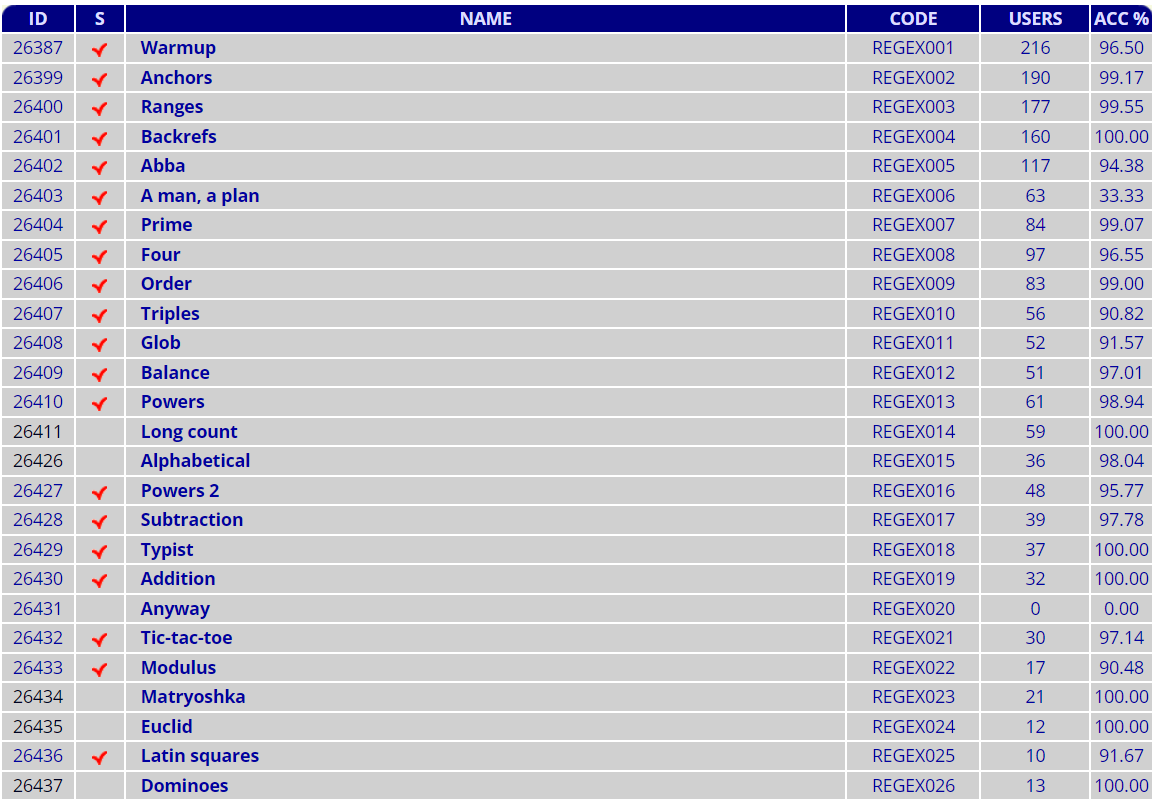
- зим - зимовье.

Регулярное выражение: \b(?:г|пгт|рп|кп|к|дп|п|п\. ст|с|д|сл|ст|ст-ца|х|у|рзд|зим)\. [А-Я][а-я]+(?:-[А-я]+)\*\b

Решение: <https://regex101.com/r/AhhdZO/1>

**Задание 8 (1 балл)**

Решить задачи (20 заданий) на ресурсе<https://www.spoj.com/SHORTEN/problems/regex/>. (Ресурс доступен через VPN. В качестве результата в отчете представить снимок экрана и пояснить логику решения.)



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | ID | Название | Регулярное выражение | Логика решения |
| 1 | 26387 | Warmup | \b.\*foo.\*\b | Последовательность символов, включающаяя «foo» |
| 2 | 26399 | Anchors | \b.\*ick\b | Последовательность символов, в конце которой обнаруживается «ick» |
| 3 | 26400 | Ranges | \b[abcdef]\*\b | Последовательность, состоящая только из символов abcdef |
| 4 | 26401 | Backrefs | \b(?<three\_letters>\w\w\w).\*(\k<three\_letters>).\*\b | Последовательность, дважды включающая в себя последовательность из трех символов |
| 5 | 26402 | Abba | \b(?!.\*(\w)(\w)\2\1).\*\b | Последовательность, в которой нет последовательности вида:  буква1 буква2 буква2 буква1 |
| 6 | 26403 | A man, a plan | \b(?<l1>\w)(?<l2>\w).\*(\k<l2>)(\k<l1>)\b | Последовательность вида:  буква1 буква2 … буква2 буква1 |
| 7 | 26404 | Prime | ^(?!(xx+)\1+$) | Последовательность из х, длина которой является простым числом |
| 8 | 26405 | Four | \b(.\*(?<vowel>[euioa]))(.\k<vowel>){3}.\*\b | Последовательность, в которой минимум 4 раза встречается одна и та же гласная буква |
| 9 | 26406 | Order | \ba\*b\*c\*d\*e\*f\*g\*h\*i\*j\*k\*l\*m\*n\*o\*p\*q\*r\*s\*t\*u\*v\*w\*x\*y\*z\*\b | Последовательность из букв, идущих в алфавитном порядке |
| 10 | 26407 | Triples | ^[0369]\*(([0369]\*([147][0369]\*[258])|((([147][0369]\*[147][0369]\*)|([258][0369]\*))((([258][0369]\*[147][0369]\*)\*[147])|([258][0369]\*([147][0369]\*[258][0369]\*)\*)[258]))[0369]\*)\*)[0369]\*$ | Последовательность, являющая числом, которое без остатка делится на 3 |
| 11 | 26408 | Glob | ^(\w\*)((?= matches \1$)|\\*(\w\*)((?= matches \1\w+\3$)|\\*(\w\*)((?= matches \1\w+\3\w+\5$)|\\*(\w\*)(?= matches \1\w+\3\w+\5\w+\7$)))).\*$ | Последовательность, в начале которой задана маска, в которой \* означает любое количество символов, а в конце приведена последовательность, соответствующая этой маске |
| 12 | 26409 | Balance | (<(?R)\*>)\* | Последовательность, в которой все скобки (< >) сбалансированы |
| 13 | 26410 | Powers | ^((((((((((x)\10?)\9?)\8?)\7?)\6?)\5?)\4?)\3?)\2?)\1?$ | Последовательность, состоящая из x, длина которой является степенью двойки |
| 14 | 26427 | Powers 2 | ^((((((((((x)(\10{2})?)(\9{2})?)(\8{2})?)(\7{2})?)(\6{2})?)(\5{2})?)(\4{2})?)(\3{2})?)(\2{2})?)(\1{2})?$ | Последовательность, состоящая из x, длина которой является степенью тройки |
| 15 | 26428 | Subtraction | ^(\w\*)(\w\*) - \1 = \2$ | Последовательность, состоящая из x, представляющая собой корректное вычитание |
| 16 | 26429 | Typist | \b(?!\w\*[yuoinml]\w\*).\*\b | Последовательность, не включающая в себя символы yuoinml |
| 17 | 26430 | Addition | ^(\w\*)(\w\*) \+ (\w\*)(\w\*) = ((\1\2 \+ \3\4)|(\1 \+ \2\3\4)|(\1\2\3 \+ \4))$ | Последовательность, состоящая из x, представляющая собой корректное сложение |
| 18 | 26432 | Tic-tac-toe | ^(.\*(\w)\2\2.\*|(\w).. \3.. \3..|.(\w). .\4. .\4.|..(\w) ..\5 ..\5|(\w).. .\6. ..\6|..(\w) .\7. \7..)$ | Последовательность вида: … … …, в которой X – ход крестика, O – ход нолика, . – пустая клетка, предтавляющая собой законченную игру в крестики-нолики |
| 19 | 26433 | Modulus | \b(((\w+)(\w\*)\4\* % \4 = (?!\4)\3)|(((\w+) % \7\w+ = \7)))\b | Последовательность, состоящая из x, представляющая собой корректный mod |
| 20 | 26436 | Latin squares | ^(\w)(\w)(\w)(\w)(?=.\*\1.\*\1.\*\1.\*)(?=.\*\2.\*\2.\*\2.\*)(?=.\*\3.\*\3.\*\3.\*)(?!.\*\1\w\w\w.\*)(?!.\*\w\2\w\w.\*)(?!.\*\w\w\3\w.\*)(?!.\*\w\w\w\4.\*) (\w)(\w)(\w)(\w)(?!.\*\5\w\w\w.\*)(?!.\*\w\6\w\w.\*)(?!.\*\w\w\7\w.\*)(?!.\*\w\w\w\8.\*) (\w)(\w)(\w)(\w)(?!.\*\9\w\w\w.\*)(?!.\*\w\10\w\w.\*)(?!.\*\w\w\11\w.\*)(?!.\*\w\w\w\12.\*).\*$ | Последовательность, состоящая только из четырех символов и пробелов, которые образуют четыре «ряда» по четыре символа, такая что ни в каком ряду и ни в каком столбце нет повторяющихся символов |

## Отработка текстов

**Задание 9 (1 балл)**

Написать программу (Python, С# или любой другой язык программирования) для построения частотного словаря текста на русском языке без учета стоп-слов. Входной текст хранится в текстовом файле, построенный словарь также должен быть сохранен в текстовый файл в формате (словоформа, количество\_вхождений\_словоформы) с сортировкой по количеству вхождений. Для деления текста на токены и удаления стоп-слов использовать библиотеку nltk, а для нормализации использовать библиотеку Pymorphy2.

**Задание 10 (1 балл)**

Реализовать вторую версию программы (см. задание 1) построения частотного словаря, в которой для нормализации используется стеммер (найти необходимую библиотеку самостоятельно). Сравнить построенные частотные словари.

**Задание 11 (1 балл)**

Собрать корпус документов по выбранной теме и по основе программ, разработанных в заданиях 1 и 2, подсчитать метрику TF-IDF для ключевых слов документа.

**Задание 12 (1 балл)**

Реализовать алгоритм автоматического реферирования (квазиреферирование) на основе статистического подхода, алгоритм приведен ниже. Входные данные: исходный текст и коэффициент сжатия. Единицей реферирования (фрагментом) должно являться предложение. Выходные данные: список ключевых слов с весами, список предложений с весами, текст реферата.

**Алгоритм**:

1. Разбить текст на предложения.

2. Разбить текст на слова произвести их нормализацию.

3. Удалить стоп-слова.

4. Подсчитать веса слов (tf или tf-idf).

5. Определить веса предложений, рассчитанный как сумма весов, входящих в предложение слов.

6. Отсортировать предложения по убыванию веса.

Решение:

Задания по обработке текстов выполнены в виде программы на языке Python. Решение представлено в репозитории <https://github.com/EkatherinaS/Data-Analysis-Technologies-in-Internet.git> в папке Lab1.