ПРАКТИЧЕСКОЕ ЗАДАНИЕ №1. ОСЕНЬ 2021

ВВЕДЕНИЕ

Регулярные выражения - мощный инструмент для обработки текстовых данных, включая тексты на естественных языках. Регулярные выражения используются в различных задачах, таких как предварительная обработка данных, системы интеллектуального анализа информации на основе правил, сопоставление с образцом, разработка текстовых функций, валидация входных данных, извлечение данных из интернета, и т. д.

ПОСТАНОВКА ЗАДАЧИ

Требуется составить регулярные выражения, для решения следующих независимых подзадач:

1. проверка корректности пароля;
2. поиск email адресов в тексте;
3. поиск URI адресов в тексте;
4. поиск дат в тексте.

1. Проверка корректности пароля

В рамках этой подзадачи требуется разработать регулярное выражение, которым возможно проверить может ли являться входная строка (целиком) корректным паролем.

Ограничения на пароли:

- пароль должен содержать только латинские символы, цифры и специальные символы:

  `!@#$%^&*?`

- пароль должен состоять из не менее чем восьми символов
- пароль должен содержать по крайней мере один латинский символ в верхнем регистре
- пароль должен содержать по крайней мере один латинский символ в нижнем регистре
- пароль должен содержать по крайней мере одну цифру
- пароль должен содержать по крайней мере два различных специальных символа
- пароль не должен содержать двух одинаковых символов подряд

ПРИМЕРЫ ПАРОЛЕЙ

корректные пароли	**некорректные пароли**
rtG3FG!Tr^e	пароль
aA1!*!1Aa	password
enroi#$rkdeR#$092uwedchf34tguv394h	qwerty
	lOngPa$$W0Rd

2. Поиск в тексте адресов электронной почты

В рамках этой подзадачи требуется разработать регулярное выражение для поиска подстрок в тексте, которые являются корректными адресами электронной почты. Адрес считается корректным, если его запись соответствует актуальному стандарту email адресов (RFC5322).

ПРИМЕР ТЕКСТА

> Please contact us at contact@example.com for further information. You can also give feedback at feedback@ex.com. Hey! @_@ We miss you, some@com!

Зеленым фоном отмечены корректные адреса электронной почты, серым – подстрока, похожая, но не являющаяся корректным адресом электронной почты.

ПРИМЕРЫ АДРЕСОВ ЭЛЕКТРОННОЙ ПОЧТЫ

корректные адреса	**некорректные адреса**
somename@example.com	example
"Xyz@func"@example.com	#%@#^$#@#$@#.com
"John Men"@example.com	Xyz\@func@example.com
order/action=social@example.com	email@1.22.333.4444.
" "@example.com	email@-example.com
example@com	just\"not\"right@example.com
$S4372@example.com	
!xyz!func%qwe@example.com	
some.name@[127.0.0.1]	

3. Поиск в тексте URI

В рамках этой подзадачи требуется разработать регулярное выражение для поиска некоторых URI в тексте. Наиболее общий стандарт URI: RFC3986

URI содержит имя протокола, необходимого для доступа к ресурсу, а также имя ресурса. Первая часть URI определяет, какой протокол использовать в качестве основного средства доступа. Вторая часть идентифицирует IP-адрес или доменное имя и, возможно, поддомен, где находится ресурс.

Регулярное выражение должно обнаруживать URI следующих протоколов:

- http (RFC7230)
- https (RFC7230)
- ftp (draft, RFC3986)
- sftp (draft)
- ssh (draft)
- smb (draft)

ПРИМЕРЫ URI

корректные URI	**некорректные URI**
http://example.com	htttttttp://example.com
https://ex.com/articles/page1?v=2&s=ex	http//example.com
http://127.0.0.1:1235	C://Users/User/example.com
ftp://example.com/files/example.txt	/home/user/example.com
http://xn--fsqu00a.xn--3lr804guic/	
ssh://login@server.com:1234/repository.git	
smb://192.168.1.7/USER$/	
ftp://ftp.example.com/path/	

4. Поиск в тексте упоминаний дат

Под упоминанием даты понимается явная текстовая отсылка к некоторому конкретному дню, или промежутку времени большего размера (неделя, месяц, год, ...). Обычно упоминание даты состоит из нескольких опциональных частей, следующих в произвольном порядке: дня недели, числа, месяца и года. Также упоминанием даты являются относительные даты (вчера, завтра и т.п.)

ПРАВИЛА ОПРЕДЕЛЕНИЯ ГРАНИЦ ДАТ:

- в упоминание даты включаются предлоги, стоящие непосредственно перед датой

> Нобелевская премия в 2017 году была получена Ивановым И. И.
>
> С августа 2019 года по декабрь 2021 года необходимо набрать…
>
> Суммарно с 2020 года по первое полугодие 2021 года было обнаружено…
>
> Этим утром его друзья отправились на рыбалку.

- в упоминание даты включаются все составные части упоминания

> В четверг, 14 июня 2021 года состоялся концерт известной группы.

- идущие подряд относительные и абсолютные упоминания дат разделяются на два упоминания

> Пять недель назад 7 августа был зафиксирован новый мировой рекорд.

- упоминания разных дат, содержащие общие слова, объединяются в одну дату

> Великий макаронный монстр являлся правителем государства с 2013 по 2020 годы

УПОМИНАНИЕМ ДАТЫ НЕ ЯВЛЯЕТСЯ:

- интервал времени, не привязанный к некоторому конкретному дню;

> На прошлой неделе истёк двухнедельный срок ультиматума
>
> А вы знали, что в прошедшем сезоне Петров обучал Иванова?
>
> Иван Иванов, в течение 12 лет занимавший первые места…

- Часть названия чего-либо

> В августе на ЧМ-2077 сборная Великой Земли одержала победу

- Относительные даты, для определения абсолютного значения которых требуются дополнительные знания о мире

> Прозрение свершилось в год тридцатилетия победы великого Ктулху!
>
> Уехал в родные края, где и жил до своих последних дней.

РЕШЕНИЕ ЗАДАЧИ

ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ

Документация на библиотеку регулярных выражений в Python3: docs.python.org/3/library/re.html

Регулярные выражения в Python. От простого к сложному: habr.com/ru/post/349860

Тестирование и отладка регулярных выражений с возможностью выбора языка программирования: regex101.com

Применение регулярных выражений для NLP: towardsdatascience.com/regex-essential-for-nlp-ee0336ef988d

ТЕСТИРОВАНИЕ

На личной странице (2021-1.tpc.ispras.ru/submissions/regexp) находится статистика со всеми результатами в т.ч. результатами последнего тестирования (дата, метрики качества).

На странице 2021-1.tpc.ispras.ru/results доступны результаты всех участников. Таблица обновляется раз в неделю.

ЗАГРУЗКА РЕШЕНИЯ

Загружаемый файл должен представлять собой ***zip архив с любым именем***. Архив должен обязательно содержать:

- Решение в файле ***solution.py***. В файле должны содержаться следующие строки, содержащие регулярные выражения:
  1. Регулярное выражение для проверки пароля на корректность (PASSWORD_REGEXP)
  2. Регулярное выражение для поиска email адресов (EMAIL_REGEXP)
  3. Регулярное выражение для поиска URI адресов (URI_REGEX)
  4. Регулярное выражение для поиска дат (DATE_REGEXP)
- Описание найденных регулярных выражений в файле description.txt. Пожалуйста, напишите подробное описание, как были найдены регулярные выражения. Это описание будет выложено вместе с решением после завершения курса.

Каждое регулярное выражение должно являться строкой, записанной по правилам python regexp. В противном случае система проверки выдаст ошибку.

ПРИМЕР РЕШЕНИЯ, ВОЗВРАЩАЮЩЕГО ПУСТЫЕ РЕЗУЛЬТАТЫ ДЛЯ ВСЕХ ПОДЗАДАЧ

```
PASSWORD_REGEXP = r''
EMAIL_REGEXP = r''
URI_REGEXP = r''
DATE_REGEXP = r''
```

ОГРАНИЧЕНИЯ

- Каждую неделю можно послать не более 25 решений.

  Внимание! Итоговое тестирование будет проводиться на последнем загруженном решении.

- Размер загружаемого архива не должен превышать 15Мб.
- Время тестирования каждого регулярного выражения не должно превышать 2 секунд на тексте из 1000 символов.
- На проверяющей машине доступно 16 Гб оперативной памяти.

Оценка качества

Для оценки задания используется усредненная F_1 мера по каждой из подзадач.

Для первой подзадачи используется F_1 мера для задачи бинарной классификации

$$P = \frac{tp}{tp + fp}; R = \frac{tp}{tp + fn}; F_1 = \frac{2PR}{P + R};$$

Для оценки остальных подзадач используется micro-averaged F_1 мера точного совпадения границ искомых подстрок:

$$P = \frac{|correct|}{|predicted|}; R = \frac{|correct|}{|expected|}; F_1 = \frac{2PR}{P + R}$$

При проверке корректности пароля, в случае превышения ограничения по времени, считается, что ответ противоположен правильному.