Лабораторная работа 1. Регулярные выражения

**Цель**. Научиться составлять регулярные выражения (РВ) и использовать их для решение следующих задач на языке C++:

1. задача поиска в строке хотя бы одной подстроки, соответствующей шаблону поиска;

2. задача поиска в строке всех подстрок, соответствующей шаблону поиска;

3. задача проверки соответствия строки шаблону;

4. задача замены в строке одной подстроки на другую.

Электронный ресурс: <https://regex101.com/>

**Задача 1**. **Составьте регулярное выражение** для определения, содержится ли в тексте время, записанное в формате hh:mm:ss (часы:минуты:секунды), где 0<=hh<=23, 0<=mm, ss<=59. Для записи часа, минут и секунд отводится по 2 знака, то есть в случае однозначного значения, слева к нему приписывается незначащий 0.

**Заполните таблицу 1.** Напишите регулярное выражение и приведите по 5 примеров, соответствующих (корректных) и несоответствующих (некорректных) паттерну.

Таблица 1. Примеры строк для РВ определения времени

|  |  |
| --- | --- |
| Регулярное выражение: ([0-1][0-9]|2[0-3]):[0-5][0-9]:[0-5][0-9] | |
| Корректные строки | Некорректные строки |
| 1. 04:25:32 | 1. 44:25:32 |
| 1. 23:25:32 | 1. 23:61:32 |
| 1. 23:59:32 | 1. 323:25:62 |
| 1. 00:25:22 | 1. 26:59:32 |
| 1. 23:59:32 | 1. 24:25:32 |

**Напишите функцию на языке C++**, которая определяет содержит ли строка время в необходимом формате. Проверьте работу функции на примерах из таблицы.

**Задача 2**. **Составьте регулярное выражение** для поиска в тексте предложений. Предложения должны начинаться с большой буквы и заканчиваться знаком препинания (. или ! или ?).

**Заполните таблицу 2.** Напишите регулярное выражение и приведите по 5 примеров, соответствующих (корректных) и несоответствующих (некорректных) паттерну.

Таблица 2. Примеры строк для РВ определения предложений

|  |  |
| --- | --- |
| Регулярное выражение [A-Z][^.!?]\*[.!?] | |
| Корректные строки | Некорректные строки |
| 1. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s. | 1. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s+ |
| 2. Lorem Ipsum has been? | 2. lorem Ipsum has been the industry's standard dummy text ever since the 1500. |
| 3. Lorem Ipsum has been! | 3. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s |
| 4. A. | 4. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, |
| 5. Lorem. | 5. Lorem2 |

**Напишите функцию на языке C++**, которая находит в тексте и сохраняет в вектор из строк все предложения.

**Задача 3**. **Составьте регулярное выражение** для имени переменной. Имя переменной – это одно непустое слово, состоящее из латинских букв любом регистре, цифр и знака подчеркивания, цифры могут идти, начиная со второй позиции. Пусть дополнительно длина имени не превышает 64 символа.

**Заполните таблицу 3.** Напишите регулярное выражение и приведите по 5 примеров, соответствующих (корректных) и несоответствующих (некорректных) паттерну.

Таблица 3. Примеры строк для РВ определения имени переменной

|  |  |
| --- | --- |
| Регулярное выражение: ^[a-zA-Z\_][a-zA-Z0-9\_]{0,63}$ | |
| Корректные строки | Некорректные строки |
| 1. myvar | 1. 2myvar |
| 2. my\_var | 2.my-var |
| 3. \_my\_var | 3. my var |
| 4. myVar | 4. my+var |
| 5. m3yvar2 | 5. !myvar |

**Напишите функцию на языке C++**, которая определяет соответствует ли строка данному паттерну, то есть является ли строка именем переменной. Проверьте работу функции на примерах из таблицы.

**Задача 4**. **Составьте регулярное выражение** для поиска некорректной записи запятых в тексте. Перед запятой не должно быть пробелов, после запятой должен быть один пробел.

**Заполните таблицу 4.** Напишите регулярное выражение и приведите по 5 примеров, соответствующих (корректных) и несоответствующих (некорректных) паттерну. В данной задаче в качестве корректных примеров можно использовать исправленные некорректные примеры.

Таблица 4. Примеры строк для РВ для расстановки запятых.

|  |  |
| --- | --- |
| Регулярное выражение (\s+,\s\*|,\s{2,}|,(?=[^\s])) | |
| Корректные строки | Некорректные строки |
| 1. abab, ababa | 1. sese,seses |
| 2. abab, ababa | 2.sseses , sesese |
| 3. abab, ababa | 3. abab, ababa |
| 4. abab, ababa | 4. sseses , sesese |
| 5. abab, ababa | 5. abab ,ababa |

**Напишите функцию на языке C++**, которая заменяет некорректную расстановку запятых в тексте на корректную. Проверьте работу функции на примерах из таблицы.

**Задача 5\***. Составить регулярное выражение для даты в формате dd.mm.yyyy (день:месяц:год), где 0<=dd<=31, 0<=mm<=12, 0<=yyyy<=2023. Для записи дня и месяца отводится по 2 знака, то есть в случае однозначного значения, слева к нему приписывается незначащий 0. Для записи года используется 4 знака, незначащие нули записываются до 4-х знаков. Регулярное выражение должно проверять корректность даты в високосный год.

**Заполните таблицу 5.** Напишите регулярное выражение и приведите по 5 примеров, соответствующих (корректных) и несоответствующих (некорректных) паттерну.

Таблица 5. Примеры строк для РВ для расстановки запятых.

|  |  |
| --- | --- |
| Регулярное выражение | |
| Корректные строки | Некорректные строки |
| 1. | 1. |
| 2. | 2. |
| 3. | 3. |
| 4. | 4. |
| 5. | 5. |

**Напишите функцию на языке C++**, которая выбирает все некорректные даты в тексте. Проверьте работу функции на примерах из таблицы.