

# Отчет

- Воспользуемся леммой о накачке. Будем рассматривать слово вида  $a^n abb^n$ , которое разобьем на три части:  $a^{n-l}$ ,  $a^l$ ,  $abb^n$ . Но тогда, если повторить середину 0 раз мы не получим слово из нашего языка. Так как  $(n-l) \neq n \rightarrow |u| \neq |v|$ . Значит, язык не является регулярным.
  - Вновь воспользуемся леммой о накачке. Если мы возьмем некоторое слово  $a^k c^m e^n$  и разделим его на три части:  $a^{k-l}$ ,  $a^l$ ,  $c^m e^n$ , тогда построив среднюю часть 0 раз, мы не получим слова из нашего языка, поскольку  $(k-l) + n + 1 < k + n + 1$  (конечно, при условии что  $l > 0$ ). Значит вновь получаем нерегулярный язык.
  - На данный момент неизвестно существует ли бесконечное число простых, для которых выполняется, что  $p, (p+2)$  - простые. Тогда рассмотрим два случая. Первый: пусть таких чисел бесконечное число, значит, для любого  $n$  найдется  $p$  такое, что  $p \geq n \wedge p, (p+2) - prime$ . Значит, наш язык это ничто иное как  $a^*$ , а значит он регулярный. С другой стороны, пусть таких чисел конечное число, а значит язык является регулярным.
- В файле solution.py представлен парсинг регулярных выражений с помощью производных на языке Python. В файле solution\_improved.py представлена улучшенная версия, в которой функция nullable возвращает true/false, а также улучшен алгоритм построения регулярных выражений. Запуск тестов происходит в файле main.py, там же находятся примеры регулярных выражений. В папке tests/ лежат тесты и выходные значения.

После проведения нескольких экспериментов, включающих оценку времени парсинга регулярных выражений до и после добавления оптимизаций, можно сказать, что время парсинга заметно уменьшилось, что наглядно видно на следующих примерах:

---

Regexp:  $(0|1|2|3|4|5)^*$

String: 01234

Answer: True

Matched time before: 0.0034

Matched time after: 0.0002

---

Regexp:  $1(01^*0)^*1$

String: 10100101

Answer: True

Matched time before: 10.8477

Matched time after: 0.0001

---

Regexp: (0|1|2|3|4|5)\*

String: 01234012345543211

Answer: True

Matched time before: –

После нескольких запусков процесс так и не смог успешно завершиться

Matched time after: 0.0004