

# Отчет

## Постановка задачи:

Реализовать несколько алгоритмов поиска подстроки в строке и сравнить их по производительности, использованию памяти.

## Параметры вычислительного узла:

Операционная система: Windows 11

Процессор: intel Core i5 -1155G7

Оперативная память: 8 гб

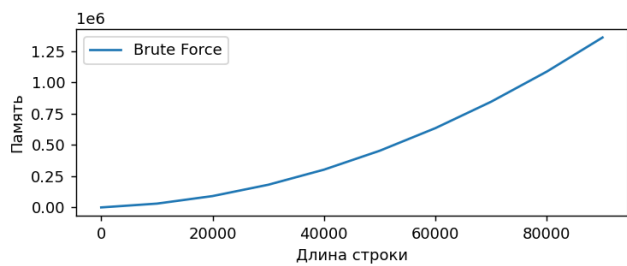
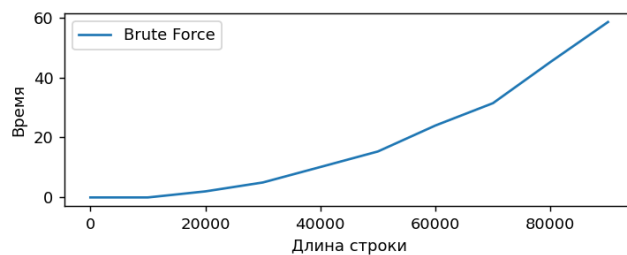
## Описание алгоритмов:

- 1) БрутФорс – метод поиска в строке подстроки напрямую, идет сравнение каждого символа подстроки с символами строки пока не произойдет совпадение.
- 2) Рабин Карп – работает с использованием хэш функции. Символы строки и подстроки переводим в хэши, считаем хэш подстроки и хэш подстроки в строке длиной равной искомой подстроке(окно), сравниваем, если хэши равны, то сравниваем сами символы.
- 3) Кнут-Моррис-Пратт – сравнивает префикс функции искомой подстроки и строки. Если значение префикс функции будет равно длине подстроки, то подстрока найдена.
- 4) Ахо-Корасик – строит для набора строк бор(дерево) в общем случае, по бору строит автомат. Затем алгоритм передает автомату все символы искомой подстроки. Под воздействием какой-то входной буквы переходим в другое состояние — т.е. в другую позицию в наборе строк. Автомат приходит в конечное состояние - строка есть в строке поиска.
- 5) Z – функция – алгоритм поиска подстроки с помощью Z - функции. Z-функция строки — это массив длины  $n$ , элемент которого равен наибольшему числу символов, начиная с позиции  $i$ , совпадающих с первыми символами

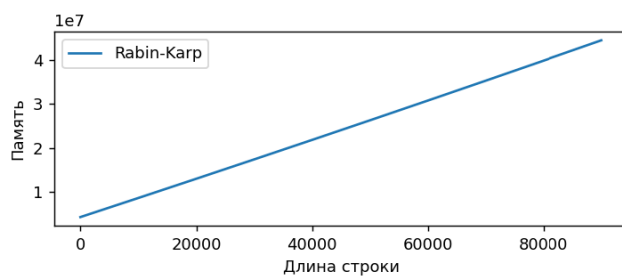
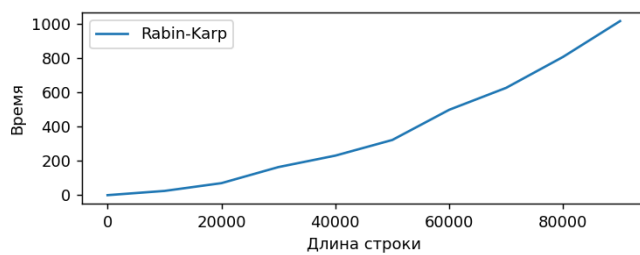
строки. Вычисляем Z-функцию от строки. Если в полученном массиве, в позициях в которых значение Z-функции равно длине искомой подстроки, то искомая подстрока найдена.

## Результаты:

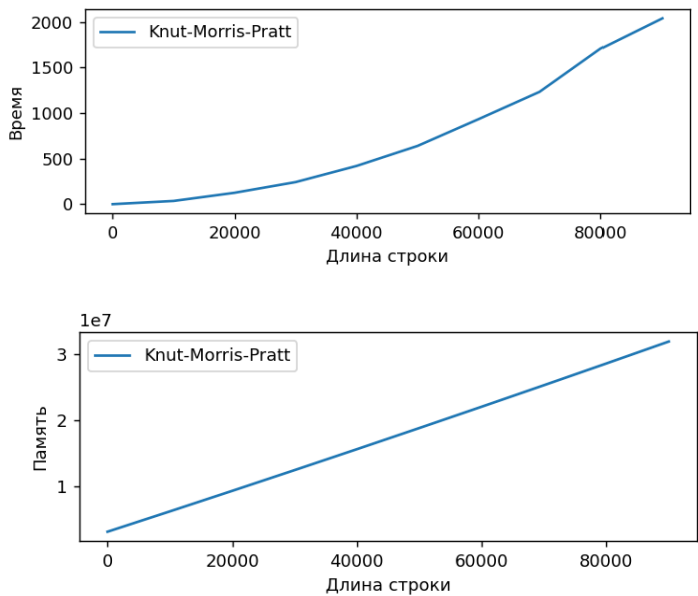
### БрутФорс



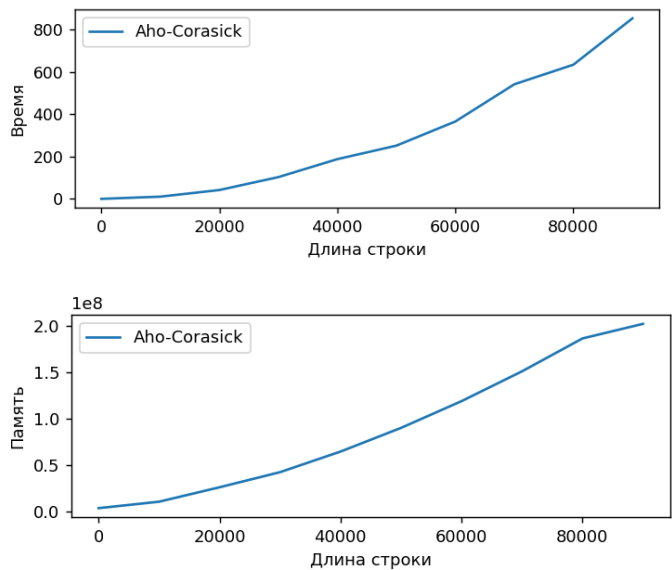
### Рабин Карп



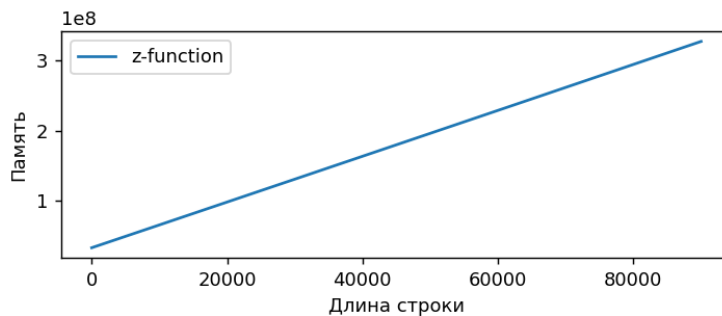
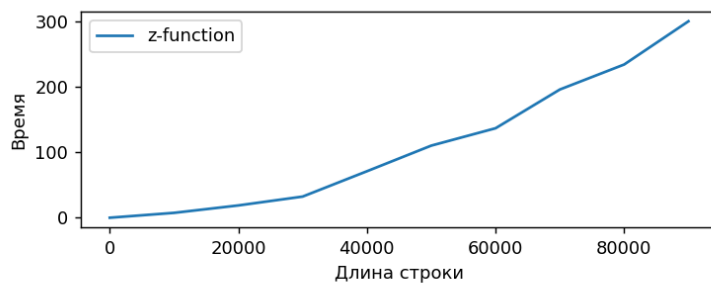
# Кнут-Моррис-Пратт



# Ахо-Корасик



## Z – функция



## Обоснование результатов:

### Сложности алгоритмов:

- брутфорс имеет квадратичную зависимость
- рабин карп, кнут-моррис-пратт, ахо-корасик и z-функция линейную зависимость

### Вывод:

Худшим алгоритмом на тестах по занимаемой памяти на больших данных оказался Брутфорс, имеющий квадратичную зависимость. Лучшими по памяти является Ахо-Корасик, но по времени он уступает другим алгоритмам. Лучшими по времени оказались алгоритмы Рабин-Карп и Брутфорс. Z-функция и алгоритм КНМ работают примерно одинаково, они оба оптимизированы, так как пропускают некоторые шаги сравнения, если уже совпадающие

символы найдены в предыдущих итерациях. На малых данных алгоритмы работают почти одинаково.