# Отчёт

Введение

Проект посвящен разработке и реализации алгоритма для поиска максимального палиндрома в заданной строке, используя метод построения дерева суффиксов по Ukkonen. Этот метод был выбран из-за его высокой эффективности и способности строить дерево суффиксов в режиме онлайн с временной и пространственной сложностью O(|T|), где T – длина строки.

Методы

Для реализации проекта был использован язык программирования C++. Был создан класс SuffixTree, который включает в себя функции для построения дерева и добавления символов строки один за другим. Активная точка, активное ребро, активная длина и оставшееся количество суффиксов были ключевыми элементами в управлении процессом добавления символов и расширения дерева.

Кроме того, в классе реализованы функции для обхода дерева и поиска узлов, содержащих суффиксы, которые представляют собой палиндромы, путем сравнения начальных и конечных индексов символов.

Результаты

Алгоритм был протестирован на различных строках, и в каждом случае он успешно строил дерево суффиксов и находил максимальный палиндром. Время выполнения алгоритма соответствовало теоретической оценке сложности O(|T|), что подтверждает его эффективность.

Выводы

Реализованный проект подтвердил, что алгоритм Ukkonen эффективен для построения деревьев суффиксов и может быть успешно применен для решения задачи поиска палиндромов. Однако процесс визуализации дерева оказался сложным из-за его размера и структуры, что стало одной из основных проблем при разработке.

Проблемы и их решения

В процессе работы над проектом возникли трудности с обработкой больших строк из-за ограничений по памяти. Это было решено оптимизацией структуры данных для узлов дерева и более эффективным управлением памятью. Также проблему представляло понимание логики работы активной точки в алгоритме Ukkonen, что требовало дополнительного времени на изучение и эксперименты.