**Аналіз часу виконання**

**1.1 Операції з множиною рядків**

* **Операція + (додавання рядка):**  
  Додавання рядка в unordered\_set виконується в середньому за час O(1) завдяки хешуванню. Однак у найгіршому випадку (при колізіях) операція може зайняти O(n), де n — кількість рядків у множині.
* **Операції – та ? аналогічно.**

Таким чином, виконання всіх операцій на множині рядків (додавання, видалення, перевірка) має середню складність:

* **Середній час виконання всіх операцій:** O(k), де k — кількість операцій.
* **У найгіршому випадку:** O(k×n).

**1.2 Пошук паліндромів**

Пошук паліндромів використовує поліноміальне хешування, що дозволяє перевіряти підрядок на паліндром за O(1). Однак перебір усіх можливих підрядків вимагає часу:

* Для кожного рядка довжиною l, пошук усіх підрядків, які можуть бути паліндромами, займе O(l^2).
* Якщо у нас N рядків, де кожен рядок має максимальну довжину l, то загальна складність алгоритму пошуку паліндромів буде O(N×l^2), де N — кількість рядків, а l — максимальна довжина рядка (в цьому випадку l≤15).

**2. Оцінка ймовірності колізій**

Для оцінки ймовірності колізій використовується парадокс днів народження. Відповідно до формули, ймовірність колізії для поліноміального хешування можна оцінити як p=1 − exp(n^2/2m), де:

* n — кількість рядків,
* m — модуль, за яким обчислюється хеш (у нашому випадку m=10^9+9).

Ймовірність колізії для поліноміального хешування при n = 10^6 і модулі m=10^9 + 9 становить практично 1. Це означає, що для такого великого числа рядків колізії майже гарантовані.