УДК 004

АНАЛІЗ ЕФЕКТИВНОСТІ АЛГОРИТМІВ ПОШУКА РЯДКА СИМВОЛІВ В ТЕКСТОВИХ НАБОРАХ РІЗНОГО ПОХОДЖЕННЯ

Кожухарь К.О., Резвіна А.С.
д.т.н., проф. каф. ІС Арсірій О.О.
Одеський Національний Політехнічний Університет, УКРАЇНА

АНОТАЦІЯ. Розглянуто алгоритми пошуку заданого рядка символів в тексті: прямий пошук, алгоритм Боєра-Мура та Кнута-Моріса-Пратта. Для отримання порівняльних характеристик ефективності алгоритмів мовою програмування C у середовищі Code :: Blocks реалізовано програмний модуль з можливістю вводу вхідних даних та представлення результатів у вигляді таблиць. Проведено ряд тестувань для визначення більш ефективного алгоритму пошуку рядка символів в текстових наборах різного типу, отримані показники ефективності представлені у вигляді графіків.

Вступ. Пошук у текстових рядках є важливою складовою при вирішенні багатьох задач, включаючи редагування тексту, пошук даних та маніпулювання ними. Швидкий пошук точно заданої послідовності символів в тексті є однією з найпростіших завдань пошуку інформації, що може бути використаний для фільтрації потенційних збігів або для пошуку пошукових термінів, які будуть виділені у вихідних даних. **Мета роботи.** Розробка та реалізація програмного комплексу з метою визначення та порівняння ефективності пошуку заданого рядка символів у текстових наборах різного походження на основі алгоритма грубої сили (прямого пошуку) та алгоритмів Кнута-Морріса-Пратта та Боєра-Мура.

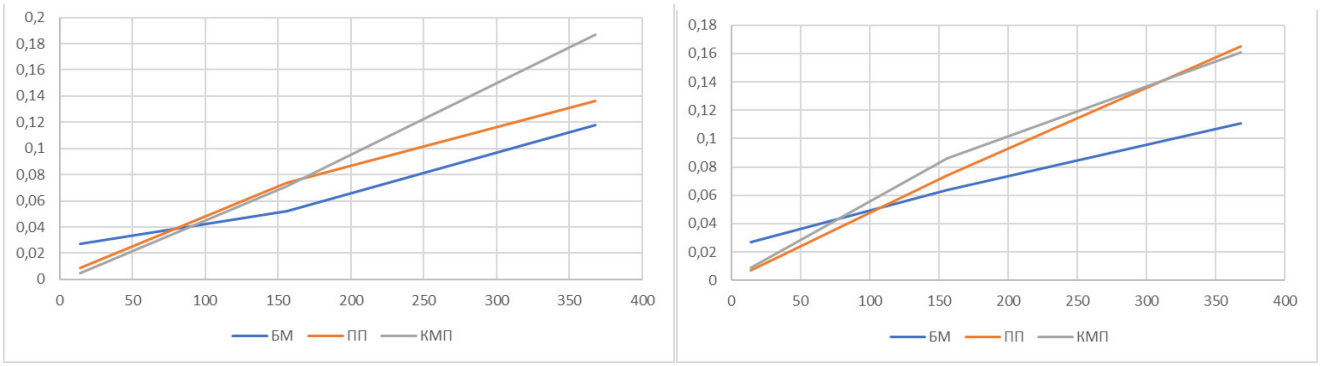
Основна частина роботи. В даний час функції пошуку заданого рядка у символьних послідовностях інкапсульовані у багатьох високорівневих мовах програмування. Проте варто пам'ятати, що стандартні функції не є завжди ефективними для послідовностей символів на різних мовах та різного походження. Також не потрібно забувати, що область застосування функцій пошуку не обмежується одними текстовими редакторами та базами даних. Алгоритми пошуку використовуються різними пошуковими роботами при індексації сторінок, і від швидкості виявлення необхідних ключових слів у тексті залежить актуальність інформації. Проблема пошуку рядків або узгодження рядків складається з пошуку всіх випадків (або першого входження) шаблону в тексті, де шаблон і текст є рядками над деяким алфавітом.

Традиційно завдання пошуку заданого рядка в наборі символів полягає у знаходженні точного входження шаблону пошуку (*needle* «голка») в рядок для пошуку (*haystack* - «стіг сіна»). При цьому задається також алфавіт Σ, на якому проводиться пошук. Для розробки програмного комплексу розглянуто теоретичні аспекти пошуку *needle* у *haystack* для поглибленого уявлення про різницю між розглянутими алгоритмами: опис основних означень порядкового пошуку, опис кожного алгоритму (включаючи опис модифікацій алгоритму Боєра-Мура) з наведенням псевдокодів та моделювання за ними. Перелік основних термінів:

1. Алгоритм Боєра-Мура - це алгоритм пошуку *needle* в *haystack*, при якому спочатку будується таблиця розміщення для іскомого *needle*, перевірка починається з останнім символом *needle* після суміщення початку *haystack* та *needle*.
2. Алгоритм прямого пошуку (грубої сили) - це алгоритм пошуку *needle* в *haystack*, при якому відбувається посимвольне порівняння *needle* та *haystack*.
3. Алгоритм Кнута-Моріса-Пратта - це лінійний алгоритм, який використовує принцип того, що кожного разу, коли відбувається збіг (або невідповідність), сам *needle* містить достатньо інформації, щоб визначити, з чого слід починати нову перевірку у *haystack*, або використовує попередню обробку *needle*.
4. Σ (Алфавіт) – кінцева множина символів
5. *Needle* - це послідовність символів пошуку (шаблон) за якою виконується пошук.
6. *Haystack* - це послідовність символів в якій виконується пошук.
7. |*haystack*| - довжина - кількість символів у рядку.
8. |*needle*| - довжина - кількість символів у шаблоні.
9. Стоп-символи – символи для подальшого зсуву *needle*
10. Префікс - це *needle*, що починається з першим символом *haystack*.

11. Суффікс - це *needle*, що закінчується останнім символом *haystack*.

З урахуванням проведеного моделювання було створено програмну реалізацію даних алгоритмів мовою програмування С у середовищі Code :: Blocks. Було визначено основні характеристики вхідних даних, за якими підібрано рядки для мануального тестування програм. Всі результати тестувань представлено у вигляді таблиць. На основі середніх значень побудовано графіки залежності часу виконання програм від кількості символів та типу Σ вхідного тексту (рисунок 1 (а, б)).



а б

Рис. 1 - Графіки залежності часу виконання програм від кількості символів та типу вхідного тексту

Графік залежності часу роботи комплексу на наборах тестових даних у вигляді зв'язного тексту наведено на рисунку 1(а). Ефективність алгоритму Боєра-Мура значно поступаєтья алгоритмам прямого пошуку та Кнута-Морріса-Пратта на короткому наборі зв'язного тексту. Ефективність алгоритму Боєра-Мура збільшується при збільшенні довжини шаблону для пошуку. Графік залежності часу роботи програм на наборах тестових даних у вигляді набору незв'язних символів наведено на рисунку 1(б). Ефективність усіх алгоритмів зменшилася, найбільший програш дає алгоритм прямого пошуку. Однак важливо відзначити, що, хоча реалізація функції пошуку за Боєром-Муром не потребує великої кількості часу, реалізація функцій препроцесінгу представляє собою доволі важку задачу. Реалізація алгоритму Кнута-Морріса-Пратта легше за реалізація алгоритму БМ, але вимагає достатнього обсягу знань та часу, так як необхідно вивчити попередньо вибраний шаблон для пошуку. У той же час реалізація алгоритму прямого пошуку проста і зрозуміла.

Висновки. Таким чином, розроблено та програмно реалізовано три алгоритми пошуку підрядка в рядку: прямий пошук, пошук за Боєром-Муром та Кнутом-Моррісом-Праттом. Важливо відзначити, що кожен алгоритм дозволяє ефективно діяти лише для свого класу завдань, про що свідчать різні вузькоспеціалізовані вдосконалення. Остаточний вибір алгоритму пошуку слід робити лише після того, як було чітко визначено поставлену задачу, визначено вхідні дані та бажані характеристики швидкості виконання задачі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Методы и алгоритмы вычислений на строках. : Пер. с англ. — М. : ООО "И.Д. Вильямс", 2006. — 496 с. : ил. — Парал. тит. англ.

2. Exact string matching algorithms - Christian Charras, Thierry Lecroq [Електронний ресурс]. – Режим доступу: http://www-igm.univ-mlv.fr/~lecroq/string/

3. String Searching Algorithms. ENG - Graham A Stephen, 1994. – 256 p.