

УДК 004.032

Мороз Н.І. – ст. гр. СН – 32

*Тернопільський національний технічний університет імені Івана Пулюя*

## **РОЗПАРАЛЕЛЕННЯ АЛГОРИТМІВ ОПРАЦЮВАННЯ СИМВОЛЬНИХ СТРІЧОК З ВИКОРИСТАННЯМ СТРУКТУРИ БОР**

Науковий керівник: д.т.н., проф. Лупенко С.А.

При розробці програмного забезпечення часто доводиться розв'язувати задачі про пошук входження однієї символічної стрічки в іншу, визначення кількості таких входжень. Перше, що спадає на думку – прямий пошук. Проходячи посимвольно по стрічці, знаходимо входження в неї першого символу іншої і перевіряємо, чи збігається кожен наступний символ. Якщо так, то розв'язок перед нами, а якщо ні – повертаємось до символу, який є наступним після першого входження. Такий алгоритм працює досить швидко при невеликому обсязі вхідних даних. Якщо ж необхідно опрацювати велику кількість стрічок тексту, то виникає необхідність використання алгоритму, який вирішить цю задачу за менший проміжок часу, не втрачаючи якості результату.

На даний час існує багато варіантів розв'язку подібних задач, але особливу увагу варто зосередити на алгоритмах, в основі яких лежить структура даних під назвою «Бор». Бор – це структура даних, призначена для зберігання набору стрічок, яка представляє собою підвішене дерево з символами на ребрах. Дещо покращеним варіантом є зжати бор, який будується за таким принципом: якщо степінь деякої вершини дорівнює одиниці, то цю вершину, ребро, яке входить в неї, і ребро, яке виходить з неї, можна об'єднати в одне ребро з більше, ніж одним символом. Такі алгоритми мають досить широку область застосування, наприклад, при роботі зі словниками чи базами даних, для пошуку документів в Інтернеті, в обчислювальній біології і навіть у системах автоматичного виправлення орфографічних помилок. Сторінки, викладені в Інтернеті, часто містять велику кількість помилок, що ускладнює їх пошук. Хотілося б, щоб такі дефекти не впливали на точність виконання пошуку. Для цього вводять наперед зафіксовану константу – максимально допустиму кількість помилок. Вона не впливає на оцінку асимптотики часу роботи алгоритму і розмірів структур даних. Описані вище алгоритми фіксують три типи помилок: пропущений символ, зайвий символ та змінений символ.

Крім високої швидкості виконання, гнучкості та точності алгоритми, в основі яких лежить бор, мають ще одну значну перевагу – зручність при розпаралелюванні алгоритму. З кожним днем все популярнішими стають паралельні обчислення, тобто обчислення, в яких кілька дій проводяться одночасно. Вони ґрунтуються на тому, що велику задачу можна розділити на кілька менших і розв'язати їх одночасно. Проблема полягає в тому, що не кожен алгоритм можна розпаралелити. Для розробки паралельних алгоритмів дуже важливим є розуміння залежностей даних. Жодна програма не може працювати швидше, ніж найдовший ланцюг залежних обчислень (відомий як критичний шлях), так як обчислення, що залежать від попередніх обчислень в ланцюгу, мають виконуватись одне за одним. Алгоритми на основі бору дають змогу без зайвих зусиль розділити задачу на необхідну кількість підзадач, що є важливим в наш час.

Таким чином величезна кількість текстової інформації може бути опрацьована за відносно невеликий проміжок часу і з високою точністю результату.