

ЛЕКЦІЯ 10.

НЕЛІНІЙНІ СТРУКТУРИ ДАНИХ

У попередніх структурах даних були реалізовані в основному одномірні відношення. Але існують структури даних, в яких виражено не тільки лінійне відношення фізичного сусідства, а значно складніші відношення між їх компонентами. Всі такі структури утворюють важливий і складний клас нелінійних структур даних, основні з яких - таблиці, дерева, спискові структури та сітки.

10.1. Таблиці

Широко розповсюдженим видом діяльності є довідково-інформаційне обслуговування, яке включає в себе накопичення даних, прийом і видачу даних на вимогу. Типовою задачею довідково-інформаційного обслуговування є задача організації і сумісного зберігання рівних записів і видачі на вимогу довільного запису незалежно від того, які записи і в якому порядку видавались раніше. В таких задачах використовують структуру даних, яку називають таблицею. Таблиця це набір поіменованих об'єктів (або записів) довільної природи, з кожним з яких однозначно пов'язане його ім'я. Ім'я запису називають його ключем. Таблиці є основною структурою запам'ятовування у файлових структурах, організованих на зовнішній пам'яті комп'ютера.

Ключ однозначно визначає місце кожного елемента в таблиці і забезпечує прямий доступ до нього. Наприклад, таблиця може мати такий вигляд:

Ідентифікатор	Адреса
(Ключ)	(Значення)
A	0101
A ₁	0102
B	0103
C	0104

Кожний запис у таблиці містить один або більше ключів. З цими ключами пов'язується тип виконуваних дій. Наприклад, ключем може бути номер службовця або його прізвище. Ключ може складатися з кількох атрибутів. Основні **властивості ключа**:

- 1) однозначна ідентифікація елемента;
- 2) відсутність надмірності.

Для одного елемента може існувати декілька наборів атрибутів, які задовольняють цим двом властивостям. Ключ, який буде використовуватися для ідентифікації записів, називають **первинним**. Для нього атрибути слід вибирати так, щоб для них був заздалегідь відомий діапазон значень, а їх кількість була б якнайменшою.

Таблиці поділяються на **впорядковані і невпорядковані**.

У найпростіших невідсортованих таблицях записи розміщуються один за одним без пропусків. Але невідсортовані таблиці неефективні при пошуку елементів, тому такі таблиці застосовують в тих випадках, коли частота доступу до їх елементів невелика.

Частіше використовуються відсортовані таблиці. Вони можуть бути відсортовані за різними атрибутами, наприклад за зростанням цифрового коду ключа або частотою звертання до записів. У першому випадку для пошуку записів застосовують алгоритм двійкового пошуку, у другому - алгоритм послідовного пошуку (див. лек. 9). Організація відсортованих таблиць вимагає додаткових витрат машинного часу.

Таблиці можуть мати ієрархічну структуру, наприклад, організовані у вигляді дерева. Структура таблиці відображає перш за все шляхи доступу при обробці індивідуальних елементів таблиці. Для всякої табличної структури характерним є те, що операції включення і виключення елементів можуть виконуватись в будь-якому місці. Операції з таблицями задаються по відношенню до ключа, а виконуються також і над записом.

Основні операції з таблицями:

- 1) включення : дана пара (ключ і запис); включити запис в таблицю так, щоб його потім можна було знайти за ключем;
- 2) заміна: дана пара (ключ і запис); знайти запис із заданим ключем і замінити його на задане значення;
- 3) виключення елемента за заданим ключем;
- 4) пошук : за заданим ключем знайти відповідний запис;
- 5) друк елементів таблиці.

У загальному випадку названі операції можна звести до двох основних: сортування і пошук. Ці операції залежать від методу доступу до елементів таблиці, а методи доступу, в свою чергу, - від структури таблиці

Існує три основних класи методів обробки таблиць в залежності від характеристики їх структур:

- 1) використовує стратегію послідовного доступу;
- 2) організовує доступ по дереву;
- 3) робить шляхи доступу до елементів таблиці безпосередніми.

10.1.1. Зображення таблиць

Структуру зберігання таблиць вибирають в залежності від типу операцій, що виконуються над ними. Для внутрішніх таблиць може бути вибрана довільна структура, яку можна побудувати в оперативній пам'яті. Залежно від методу доступу до елементів таблиці поділяються на **послідовні, деревоподібні і з обчислювальними адресами**. Останні, в свою чергу, поділяються на таблиці **з прямим доступом і хеш-таблиці**. Всі, крім таблиць з прямим доступом, можна зберігати у векторній і зчепленій пам'яті.

У найпростіших послідовних таблиць для зберігання одного запису потрібно щонайменше одну ділянку пам'яті з двома полями: КЛЮЧ і значення. Таблиця може зображуватися як вектор із цих ділянок, або як лінійний список.

Деревоподібні таблиці або таблиці, організовані як дерева порівнянь, також можуть бути зображені у векторній і зчепленій пам'яті. У першому випадку елементи даних повинні бути відсортовані за значенням ключа, у другому - зображені у вигляді двозв'язного списку. Для цього до кожного елемента таблиці повинно бути приєднано два вказівники - відповідно на ліве і праве піддерево. Тоді елемент таблиці перетворюється до такого:

ЛВ	КЛЮЧ	ТІЛО	ПВ
----	------	------	----

де ЛВ, ПВ - відповідно лівий і правий вказівники.