

РОЗДІЛ 1

ЛІНІЙНІ СТРУКТУРИ ДАНИХ ТА ЇХ ОРГАНІЗАЦІЯ В ЕОМ

Поняття лінійних структур даних. Лінійні структури із послідовним розташуванням в пам'яті ЕОМ. Масиви. Адресація елементів масиву. Стеки. Алгоритми включення та видалення елементів. Інші операції над стеками. Черги звичайні і черги кругові. Алгоритми включення та видалення елементів із черг.

Лінійні структури даних – це такі структури, у яких сусідні елементи логічно розташовані поруч так, що, знаючи один елемент, можна знайти сусідній (попередній або/і наступний). В пам'яті ЕОМ вони можуть бути розташованими теж поруч (послідовна організація або послідовне розташування), а можуть бути розташовані в різних ділянках пам'яті (зв'язане розташування).

Структури даних поділяють на примітивні та не примітивні. Примітивні – це числа (цілі, дійсні), символи, вказівники. Із символів утворюються більш складні структури – рядки, які були розглянуті у попередньому розділі. До непримітивних лінійних структур відносять пакети, масиви, списки, стеки, черги.

Пакет є, у загальному випадку, неупорядкованою множиною об'єктів, який має фіксований або змінний розмір. Вони використовуються в задачах обробки даних. Зокрема, таким структурами можна вважати файли.

Масив – це упорядкована множина, що складається з фіксованого (для статичних масивів) або не фіксованого (для динамічних масивів) числа однотипних об'єктів. Над масивами не існує операцій додавання та видалення без зсуву елементів, але інколи такі операції розглядаються і для таких структур. Масив являє собою лінійну структуру з послідовним (лінійним) розташуванням елементів у пам'яті. Алгоритми роботи з масивами приклади їх реалізації розглядаються у п. 1.1.

Список – це упорядкована множина, що складається з певного числа елементів, до котрих застосовуються операції додавання та видалення. Список, що відображає відношення сусідства між елементами, називається лінійним. Список являє собою лінійну структуру із зв'язаним розташуванням елементів у пам'яті. Алгоритми роботи зі списками розглядаються у п. 1.6.

Структури та організація даних в ЕОМ

Визначення стека і черги буде наведено нижче. Вони можуть бути реалізовані як із використанням масивів (п. 1.7), так і за допомогою списків (п. 1.8), тобто вони можуть мати як послідовну структуру зберігання, так і зв'язану.

Найбільш типовими операціями, які потребують окремого програмування, над непримітивними структурами даних вважаються:

- створення іменованої структури даних (*create*);
- видалення іменованої структури даних (*drop, erase*);
- зміна внутрішньої структури самої іменованої структури даних (*alter, modify*);
- копіювання або переміщення всієї іменованої структури даних (*copy* або *move*);
- об'єднання кількох однорідних іменованих структур даних в одну (*union, append*);
- упорядкування (сортування) за певними критеріями (*sort, order by*).

Найбільш типовими операціями над елементами неprimітивних структур даних вважаються:

- пошук елемента за певним критерієм (*select*);
- додавання елемента (*insert*);
- видалення елемента (*delete*);
- оновлення елемента (*update*);
- визначення числа елементів (*count*);
- визначення суми значень елементів (*sum*);
- визначення максимального, мінімального або середнього арифметичного значення елементів (*maximum (max), minimum (min), average (avg)*).

Для послідовних структур збереження даних використовують три основних методи пошуку (доступу) елементів:

- *послідовний* (сканування та перевірка значень елементів);
- *прямий*, коли його адреса зберігається в пам'яті (в багатьох мовах програмування така адресація може бути реалізована, наприклад, вказівниками) або обчислюється за допомогою певних функцій;
- *комбінований*, при якому за певними правилами використовуються обидва методи.

Послідовний метод пошуку є тривіальним: лічильник (індекс) елементів, починаючи, як правило з першого, збільшується на одиницю, зчитується значення потрібного елемента, яке порівнюється із шуканим. Розглянемо структури, коли адреса обчислюється. Існує багато структур

Фісун М.Т., Цибенко Б.О.

даних, що припускають звернення до будь-якого елемента на підставі його позиції в структурі. Введемо поняття функції адресації. Функція адресації для структури даних, що містить n елементів, є функцією, котра i -му елемента структури даних ставить у відповідність число, що знаходиться між 1 и n . Якщо $f(i) = a + b * i$, це лінійна функція адресації, то структура даних відноситься до лінійних з послідовною організацією.