

Struktur Data

Berikut adalah beberapa struktur data umum yang digunakan dalam pemrograman:

1. **Array**: Kumpulan elemen data yang disusun secara berurutan dan diakses menggunakan indeks.
2. **Linked List**: Kumpulan node yang setiap node mengandung data dan referensi ke node berikutnya dalam urutan.
3. **Stack**: Struktur data LIFO (Last In, First Out) di mana elemen yang terakhir dimasukkan adalah yang pertama dikeluarkan.
4. **Queue**: Struktur data FIFO (First In, First Out) di mana elemen yang pertama dimasukkan adalah yang pertama dikeluarkan.
5. **Heap**: Pohon biner khusus yang memenuhi properti tertentu (misalnya, heap maksimum atau minimum) yang sering digunakan untuk implementasi priority queue.
6. **Hash Table**: Struktur data yang mengimplementasikan associative array atau mapping dari kunci ke nilai menggunakan fungsi hash untuk menghitung indeks dari kunci.
7. **Tree**: Struktur data hierarkis yang terdiri dari simpul yang terhubung dengan tepi.
 - **Binary Tree**: Pohon dengan setiap simpul memiliki maksimal dua anak.
 - **Binary Search Tree (BST)**: Pohon biner di mana setiap simpul kiri memiliki nilai yang lebih kecil daripada simpulnya, dan setiap simpul kanan memiliki nilai yang lebih besar.
 - **AVL Tree**: Jenis pohon biner yang seimbang di mana perbedaan tinggi subtree kiri dan kanan dari setiap simpul tidak lebih dari satu.
 - **Red-Black Tree**: Jenis pohon biner yang seimbang dengan aturan spesifik yang mempertahankan keseimbangan selama operasi insert dan delete.
8. **Graph**: Kumpulan node (atau verteks) yang terhubung oleh sisi (atau edge). Graf bisa tidak berarah atau berarah, dan dapat memiliki bobot pada sisi (weighted graph).
9. **Trie**: Struktur data khusus untuk menyimpan koleksi string yang memungkinkan pencarian string dalam waktu linier.
10. **Set**: Kumpulan elemen unik yang tidak diurutkan.
11. **Priority Queue**: Antrian di mana setiap elemen memiliki prioritas terkait dengannya.
12. **Deque (Double-Ended Queue)**: Antrian di mana elemen dapat dimasukkan atau dikeluarkan dari kedua ujung.
13. **Sparse Matrix**: Representasi data yang menghemat memori untuk matriks dengan banyak elemen nol.
14. **Union-Find (Disjoint Set)**: Struktur data yang memelihara kumpulan disjoint dari elemen, berguna untuk menangani operasi gabungan dan pencarian dalam permasalahan graf.
15. **Bloom Filter**: Struktur data probabilistik yang digunakan untuk memeriksa keanggotaan suatu elemen dalam sebuah set.

Setiap struktur data memiliki karakteristik dan kegunaan yang berbeda, dan pemilihan yang tepat sangat tergantung pada jenis data dan operasi yang diperlukan dalam aplikasi atau algoritma yang

sedang dikembangkan.

Revision #1

Created 13 December 2024 11:51:51 by Admin

Updated 15 December 2024 02:52:01 by Admin