



ANALISIS PENERAPAN STRUKTUR DATA DALAM MENINGKATKAN EFISIENSI PENGEMBANGAN PERANGKAT LUNAK

Rida Ramli, Nurzyakila, Arabela, Nirmala Dwi Astuti,

Akbar, Ketring Rinayanti Manullang

Program Studi Pendidikan Teknologi Informasi, Universitas Sulawesi Barat

Abstrak

Latar Belakang: Struktur data merupakan komponen penting dalam ilmu komputer yang digunakan untuk mengatur, menyimpan, dan mengelola data secara efisien. Pemilihan struktur data yang tepat dapat meningkatkan kinerja algoritma dan efisiensi penggunaan memori dalam suatu sistem. Penelitian ini bertujuan untuk menganalisis penerapan berbagai jenis struktur data dalam pengembangan perangkat lunak. Metode yang digunakan adalah studi literatur dengan membandingkan beberapa jenis struktur data seperti array, linked list, stack, queue, dan tree. Hasil penelitian menunjukkan bahwa setiap struktur data memiliki kelebihan dan kekurangan tergantung pada kebutuhan sistem. Oleh karena itu, pemilihan struktur data harus disesuaikan dengan karakteristik permasalahan yang dihadapi.

Kata Kunci: struktur data, algoritma, efisiensi, perangkat lunak.

PENDAHULUAN

Perkembangan teknologi informasi yang pesat menuntut sistem komputer untuk bekerja secara cepat dan efisien. Salah satu faktor penting yang mempengaruhi kinerja sistem adalah penggunaan struktur data yang tepat. Struktur data memungkinkan data disimpan dan diakses dengan cara yang optimal.

Dalam pengembangan perangkat lunak, kesalahan dalam memilih struktur data dapat menyebabkan penurunan performa sistem. Oleh karena itu, pemahaman tentang berbagai jenis struktur data sangat diperlukan oleh programmer dan pengembang sistem.

TINJAUAN PUSTAKA

1. Pengertian Struktur

Data

Struktur data adalah cara mengorganisasi dan menyimpan data sehingga dapat digunakan secara efisien.

2. Jenis-Jenis Struktur Data

Berikut beberapa jenis struktur data yang umum digunakan:

- Array: Struktur data yang menyimpan elemen dengan tipe yang sama dalam satu lokasi memori berurutan.
- Linked List: Struktur data yang terdiri dari node yang saling terhubung.
- Stack: Menggunakan prinsip LIFO (Last In First Out).
- Queue: Menggunakan prinsip FIFO (First In First Out).
- Tree: Struktur data berbentuk hierarki.
- Graph: Digunakan untuk merepresentasikan hubungan antar data.

METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah metode kualitatif dengan pendekatan studi literatur. Data dikumpulkan dari buku, jurnal, dan sumber terpercaya lainnya yang berkaitan dengan struktur data.

Pembahasan

1. Penerapan Struktur

Data

- Array digunakan dalam penyimpanan data statis seperti daftar nilai mahasiswa.
- Stack digunakan pada fitur undo/redo dalam aplikasi.
- Queue digunakan pada sistem antrian seperti layanan pelanggan.

- Tree digunakan dalam sistem database dan pencarian data.

2. Analisis Efisiensi

Efisiensi struktur data dapat dilihat dari:

- Waktu akses data
- Penggunaan memori
- Kompleksitas algoritma

Pemilihan struktur data yang tepat dapat mengurangi kompleksitas waktu (time complexity) dan meningkatkan performa sistem.

KESIMPULAN

Struktur data memiliki peran penting dalam pengembangan perangkat lunak. Setiap jenis struktur data memiliki karakteristik yang berbeda, sehingga penggunaannya harus disesuaikan dengan kebutuhan sistem. Pemilihan struktur data yang tepat dapat meningkatkan efisiensi dan kinerja aplikasi secara keseluruhan.

DAFTAR PUSTAKA

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.). MIT Press.

Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). *Data Structures and Algorithms in Python*. Wiley.

Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4th ed.). Addison-Wesley.

Weiss, M. A. (2013). *Data Structures and Algorithm Analysis in C++* (4th ed.). Pearson.

Tanenbaum, A. S., & Austin, T. (2012). *Structured Computer Organization*. Pearson.

Knuth, D. E. (1997). *The Art of Computer Programming, Volume 1: Fundamental Algorithms*. Addison-Wesley.

Sahni, S. (2005). *Data Structures, Algorithms, and Applications in Java*. McGraw-Hill.

Lafore, R. (2002). *Data Structures and Algorithms in Java*. Sams Publishing.

Wirth, N. (1976). *Algorithms + Data Structures = Programs*. Prentice Hall.

Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach*. McGraw-Hill.

Nugroho, A. (2018). *Struktur Data dan Algoritma dengan Python*. Andi Publisher.

Kadir, A. (2014). *Dasar Pemrograman dan Struktur Data*. Andi Publisher.