

1. Sebuah kelas berisi 40 mahasiswa tetap dari awal sampai akhir semester. Jelaskan mengapa penggunaan array lebih efisien dibandingkan membuat 40 variabel terpisah untuk menyimpan NIM mahasiswa.

penggunaan array efisien untuk menyimpan NIM mahasiswa secara terpisah karena array mampu untuk mengelompokkan tipe data jadi satu yang disini adalah NIM yang dimana bisa digunakan string, sebab banyaknya data dan terdapat perbedaan menyeluruh pada angka NIM, jadi bisa dibuat dengan string yang mampu mengatur, mengurutkan, dan memudahkan juga jika ingin mencari NIM itu kembali.

2. Jika seorang mahasiswa pindahan ingin masuk di urutan ke-10, apa yang harus dilakukan oleh program ketika menggunakan array? Mengapa proses ini dianggap kurang efisien dibandingkan hanya mengganti nilai di posisi tertentu?

Jika ada mahasiswa pindahan yang masuk di urutan ke-10, program dengan array harus menggeser semua elemen dari posisi ke-10 hingga akhir agar bisa memberi ruang bagi data baru. Misalnya, dari total 40 mahasiswa, sebanyak 30 data harus digeser ke kanan. Proses ini dianggap kurang efisien karena memakan waktu dan sumber daya lebih banyak dibandingkan sekadar mengganti nilai pada posisi tertentu. Array bersifat statis dan menyimpan data dalam memori berurutan, sehingga penyisipan di tengah membutuhkan penggeseran elemen lain.

3. Dalam kelas yang jumlah pesertanya sering berubah karena ada mahasiswa yang drop atau add mata kuliah, akan lebih baik menggunakan Linked List. Jelaskan satu kelebihan Linked List dalam kasus ini, sekaligus satu kelemahannya dibandingkan array.

Jika jumlah peserta kelas sering berubah akibat ada mahasiswa yang menambah atau mengurangi mata kuliah, penggunaan linked list lebih baik dibandingkan array. Kelebihan linked list adalah sifatnya yang dinamis, sehingga elemen dapat ditambahkan atau dihapus di posisi mana pun tanpa perlu menggeser seluruh data. Namun kelemahannya, linked list tidak memungkinkan akses langsung ke elemen tertentu. Untuk mengakses data di urutan ke-20, misalnya, program harus menelusuri node satu per satu dari awal hingga elemen tersebut ditemukan.

4. Sebuah sistem absensi online ingin tetap cepat dalam mengakses mahasiswa tertentu, tetapi juga fleksibel ketika ada penambahan peserta baru di akhir. Mengapa vector lebih cocok dipakai dibandingkan array biasa?

Dalam sistem absensi online yang membutuhkan akses cepat ke data mahasiswa tertentu tetapi juga fleksibel dalam penambahan peserta baru, vector lebih cocok digunakan daripada array biasa. Vector memiliki kelebihan yang sama seperti array, yaitu dapat mengakses elemen secara langsung dengan indeks. Namun vector lebih fleksibel karena ukurannya dapat bertambah secara otomatis saat ada data baru. Dengan demikian, vector lebih sesuai untuk kebutuhan absensi online yang menuntut kecepatan sekaligus fleksibilitas.

5. Bayangkan kamu sedang diminta memilih struktur data untuk membuat aplikasi daftar peserta ujian:

- Peserta awal = 30 orang.
- Bisa saja ada tambahan peserta baru.
- Data perlu sering diakses berdasarkan nomor urut.
- Struktur data mana yang lebih tepat (array, linked list, atau vector) dan mengapa?

Untuk aplikasi daftar peserta ujian dengan jumlah awal 30 orang, tetapi memungkinkan adanya peserta tambahan, serta memerlukan akses data berdasarkan nomor urut, struktur data yang paling tepat adalah vector. Array tidak cocok karena ukurannya kaku, sedangkan linked list tidak efisien untuk akses berdasarkan indeks karena harus ditelusuri satu per satu. Vector memberikan fleksibilitas dalam menambah peserta baru dan tetap memungkinkan akses cepat berdasarkan nomor urut, sehingga menjadi pilihan yang paling sesuai.

## DAFTAR PUSTAKA

*Erkamim, M., Abdurrohim, I., Yuliyanti, S., Karim, R., Rahman, A., Admira, T.M.A., & Ridwan, A. 2024. Buku Ajar Algoritma dan Struktur Data. Jambi: PT Sonpedia Publishing Indonesia.*

*Ginting, S.H.N., Effendi, H., Kumar, S., Marsisno, W., Sitanggang, Y.R.U., Anwar, K., Santiari, N.P.L., Setyowibowo, S., Sigar, T.R., Atho'illah, I., Setyantoro, D., & Smrti, N.N.E. 2023. Pengantar Struktur Data. Medan: PT Mifandi Mandiri Digital.*

*Putri, M.P., Barovih, G., Azdy, R.A., Yuniansyah, Saputra, A., Sriyeni, Y., Rini, A., & Admojo, F.T. 2022. Algoritma dan Struktur Data. Bandung: Widina Bhakti Persada.*