# Soal Pertemuan 9

## Nama : Adiweno Rashad Sahita

## NIM : 15241032

## Kelas : 15.1D.07

**Soal 1:**

Jelaskan pengertian dan fungsi dari larik (array) dalam pemrograman. Berikan contoh implementasi larik dalam bahasa pemrograman pilihan anda, serta jelaskan bagaimana larik dapat digunakan untuk menyelesaikan masalah tertentu.

**Jawab:**

Larik adalah jenis variabel yang dapat digunakan untuk menyimpan sejumlah data dengan tipe yang sama. Contoh:

a = 1, 2, 3, 4, 5

print(a[2])

#output:

#3

Larik bisa membuat lebih efisien dalam penggunaan variabel dan mempermudah dalam pekerjaan dalam penyimpanan dan pemrosesan data.

**Soal 2:**

Bagaimana cara kerja indeks dalam larik (array)? Jelaskan mengapa indeks dimulai dari 0 dalam kebanyakan bahasa pemrograman dan bagaimana pengaruhnya terhadap pengelolaan memori dan kinerja algoritma.

**Jawab:**

Misal larik dari integer [1, 2, 3], indeks akan dimulai dari 0 sampai n-1, jika dipanggil indeks 0 maka outputnya adalah 1, indeks 2 maka outputnya adalah 3, dan seterusnya. Indeks dimulai dari 0 karena mesin komputer memiliki memori dan data dalam memori dimulai dari 0, jika indeks dimulai dari 1 maka komputer harus menambah memori bahwa indeks 0 adalah 1 sehingga membuat kinerja tidak efisien.

**Soal 3:**

Sebuah algoritma memiliki larik yang berisi 10 angka. Buatlah algoritma untuk mencari nilai maksimum dan minimum dalam larik tersebut tanpa menggunakan fungsi bawaan. Jelaskan bagaiman algoritma ini bekerja langkah demi langkah dan analisis kompleksitas waktu yang dimilikinya.

**Jawab:**

a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

for i in range(len(a)):

if i == 0:

print(a[i], "adalah nilai awal.")

if i == len(a) - 1:

print(a[i], "adalah nilai akhir.")

#output

#1 adalah nilai awal.

#10 adalah nilai akhir.

- a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] adalah variabel a dengan indeks 1 sampai 10.

- for i in range(len(a)): untuk mengetahui panjang/jumlah value dalam variabel a yaitu 10 value dan membuat pengulangan dari jarak indeks 0 hingga 9 dari variabel a kedalam variabel i.

- if i == 0: jika variabel i (indeks 0) sama dengan 0 maka outputnya adalah “1 adalah nilai awal” dari print(a[i], "adalah nilai awal.")

- if i == a[a] - 1: jika variabel i (indeks 9) sama dengan jumlah value a-1 (10-1= 9) maka outputnya adalah “10 adalah nilai awal” dari print(a[i], "adalah nilai akhir.")

**Soal 4:**

Dalam pemrograman, sering kali kita menggunakan larik satu dimensi. Jelaskan perbedaan antara larik satu dimensi dan larik dua dimensi, serta berikan contoh aplikasi di mana larik dua dimensi lebih efektif digunakan dibandingkan larik satu dimensi.

**Jawab:**

Larik satu dimensi adalah larik yang hanya berisi sebuah larik dalam satu variabel sementara larik dua dimensi adalah sebuah larik yang juga berisi sebuah larik lagi (bercabang). Larik dua dimensi akan lebih efektif jika memiliki dua variabel, sebagai contoh:

- Larik satu dimensi

merk\_mobil = ["Toyota", "Ford"]

merk\_motor = ["Honda", "Suzuki"]

print(f"Merk mobil: {merk\_mobil} dan merk motor: {merk\_motor}")

- Larik dua dimensi

kendaraan = [["Toyota", "Ford"], ["Honda", "Suzuki"]]

print(f"Merk mobil: {kendaraan[0]} dan merk motor: {kendaraan[1]}")

Bisa dilihat kalau larik dua dimensi dapat mengerjakan dua larik sekaligus dalam satu variabel.

**Soal 5:**

Tulis dan jelaskan algoritma untuk membalikkan urutan elemen dalam sebuah larik. Setelah itu, evaluasi algoritma tersebut dari sisi efisiensi dalam hal waktu dan ruang (space complexity).

**Jawab:**

array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

array.reverse()

print(array)

- Secara efisiensi waktu, fungsi .reverse() langsung memodifikasi daftar asli tanpa membuat salinan. Oleh karena itu, operasi ini hanya perlu membalikkan elemen dalam array dengan mengakses setiap elemen setidaknya satu kali.

- Secara efisiensi ruang, fungsi .reverse() tidak ada struktur data tambahan yang dibuat untuk menyimpan salinan array. Operasi ini hanya membutuhkan ruang untuk variabel sementara untuk menukar elemen.