

LAPORAN  
PRAKTIKUM INFORMASI DAN STRUKTUR DATA

SEMESTER GANJIL 2022/2023

MODUL 5

*Pointer*



NAMA : ALEXIS DIVASONDA SIGAT NGAING

NPM : 210711407

PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ATMA JAYA YOGYAKARTA  
TAHUN 2022

Membuat program untuk mengakses data tertentu menggunakan pointer. Indeks diakses menggunakan matriks. Data ditampilkan pada program dengan bentuk seperti lemari berpintu 20.

Menu pertama adalah memilih posisi, dengan tiga submenu di dalamnya. Submenu pertama berfungsi untuk memindahkan posisi sesuai lokasi ID yang diinginkan. ID akan dicari dengan fungsi `Peserta* cariID(Peserta* ptrDisplay, Peserta peserta[ROW][COL], int id)` dimana fungsi ini akan mengecek di dalam matriks apakah ID yang dimaksud ada. Terdapat `Peserta p = makePeserta(0, "-", "-", "-", "?", "-");` yang akan digunakan jika ID tidak ditemukan. Jika ID yang dicari ada, maka fungsi akan mengembalikan `return ptrCopy;` dan jika ID yang dicari adalah ID yang saat ini ditunjuk oleh pointer maka fungsi akan mengembalikan `return ptrDisplay;` dan jika ID tidak ditemukan maka akan mengembalikan `return temp;`. Pada submenu kedua, yaitu geser kanan, pointer akan digeser ke kanan atau ditambah dengan `ptrDisplay++;`. Submenu ketiga berfungsi untuk menggeser pointer ke kiri dengan `ptrDisplay--;` atau dalam kata lain mengurangi pointer yang diunjuk.

Menu kedua adalah isi data. Data akan diinputkan seperti biasa dan langsung tersimpan di alamat pointer yang sedang ditunjuk.

Menu ketiga adalah menu ubah data. Mirip seperti menu kedua, namun di menu ketiga ini tidak dapat dilakukan edit data jika pada alamat atau pointer saat ini kosong.

Menu keempat adalah menu hapus data. Data yang dihapus adalah data pada alamat atau pointer yang sedang ditunjuk. Program akan meminta konfirmasi terlebih dahulu. Data akan dihapus secara langsung di main program selayaknya melakukan inisialisasi. Salah satu contohnya adalah `strcpy(ptrDisplay->nama, "-");`.

Menu kelima berfungsi untuk memindahkan data pada alamat saat ini ke alamat yang berbeda. Terdapat variabel `oldID = ptrDisplay->id;` yang merupakan ID saat ini dan `newID = id;` yang merupakan ID baru yang dituju.

Kemudian terdapat Peserta \*temp = cariID(ptrDisplay, peserta, oldID); dimana menggunakan ID lama pada fungsi cariID-nya yang selanjutnya diinisialisasikan sebagai \*temp. Alamat atau pointer akan berpindah ke alamat ID baru dan data akan diinputkan sesuai data pada alamat ID sebelumnya dengan kode \*ptrDisplay = makePeserta(newID, temp->no\_urut, temp->nama, temp->daftar, temp->gender, temp->status); dan data pada alamat ID lama akan dihapus dengan \*ptrDisplay = makePeserta(oldID, "-", "-", "-", "?", "-");.