

Tugas Besar Algoritma Pemograman

Kelompok 22 – Kantin Tel - U

Nabiel Muhamad Irfani 103032330140

Muhammad Nabil Alfarizi 103032300117

[Arkbil/Tubes-Alpro \(github.com\)](https://github.com/Arkbil/Tubes-Alpro)

```
package main

import (
    "fmt"
    "os"
    "os/exec"
)

const NMAX int = 10
const HARGASEWA int = 200000

type Makanan struct {
    NamaMakanan string
    HargaMakanan int
    Stok         int
}

type Tenant struct {
    NamaTenant      string
    DataMakanan     [NMAX]Makanan
    JumlahMakanan   int
    TotalTransaksi  int
}

var tenants [NMAX]Tenant
var jumlahTenant int
var totalAdmin int
var totalSewaAdmin int

func main() {
    var pilih int
    intro()
    for {
        menu_utama(&pilih)
```

```

        switch pilih {
        case 1:
            menu_tambahkan_data_tenant()
        case 2:
            menu_mengubah_data_tenant()
        case 3:
            menu_hapus_data_tenant()
        case 4:
            transaksi_data_tenant()
        case 5:
            tampilkan_data_tenant()
        case 6:
            urutkan_data_tenant()
        case 7:
            cari_tenant()
        default:
            clear_screen()
        }
        if pilih == 8 {
            break
        }
    }
    bye()
}

func intro() {
    clear_screen()
    fmt.Println("Selamat datang")
}

func bye() {
    clear_screen()
    fmt.Println("Sampai jumpa")
}

func clear_screen() {
    c := exec.Command("clear")
    c.Stdout = os.Stdout
    c.Run()
}

func menu_utama(p *int) {
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
}

```

```

    fmt.Println("1. Tambahkan Data Tenant    ")
    fmt.Println("2. Ubah Data Tenant          ")
    fmt.Println("3. Hapus Data Tenant                    ")
    fmt.Println("4. Transaksi Tenant                      ")
    fmt.Println("5. Tampilkan Tenant                     ")
    fmt.Println("6. Urutkan Data Tenant                   ")
    fmt.Println("7. Cari Tenant                          ")
    fmt.Println("8. Exit                                ")
    fmt.Println("-----")
    fmt.Print("Pilih (1/2/3/4/5/6/7/8): ")
    fmt.Scan(p)
}

func menu_tambahkan_data_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Tambahkan Tenant        ")
    fmt.Println("2. Tambahkan Data Tenant   ")
    fmt.Println("3. Exit                     ")
    fmt.Println("-----")
    var pilih int
    fmt.Print("Pilih (1/2/3): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        menambahkanTenant()
    case 2:
        menambahkanDataTenant()
    case 3:
        return
    }
}

func menambahkanTenant() {
    if jumlahTenant >= NMAX {
        fmt.Println("Tidak bisa menambah tenant lagi. Kapasitas penuh.")
        return
    }
    fmt.Print("Masukkan nama tenant: ")
    fmt.Scan(&tenants[jumlahTenant].NamaTenant)
    tenants[jumlahTenant].JumlahMakanan = 0
    tenants[jumlahTenant].TotalTransaksi = 0
    jumlahTenant++
}

```

```

    totalSewaAdmin = totalSewaAdmin + HARGASEWA
    menu_tambahkan_data_tenant()
}

func menambahkanDataTenant() {
    var pilihTenant int
    // << sequential search >>
    fmt.Println("Daftar Tenant:")
    for i := 0; i < jumlahTenant; i++ {
        fmt.Printf("%d. %s\n", i+1, tenants[i].NamaTenant)
    }
    fmt.Print("Pilih tenant (1/2/3/...): ")
    fmt.Scan(&pilihTenant)
    if pilihTenant < 1 || pilihTenant > jumlahTenant {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihTenant--

    if tenants[pilihTenant].JumlahMakanan >= NMAX {
        fmt.Println("Tidak bisa menambah data makanan lagi. Kapasitas penuh.")
        return
    }
    var makanan Makanan
    fmt.Print("Masukkan nama makanan: ")
    fmt.Scan(&makanan.NamaMakanan)
    fmt.Print("Masukkan harga makanan: ")
    fmt.Scan(&makanan.HargaMakanan)
    fmt.Print("Masukkan stok makanan: ")
    fmt.Scan(&makanan.Stok)
    tenants[pilihTenant].DataMakanan[tenants[pilihTenant].JumlahMakanan] =
makanan
    tenants[pilihTenant].JumlahMakanan++
    menu_tambahkan_data_tenant()
}

func menu_mengubah_data_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Ubah Data Tenant      ")
    fmt.Println("2. Exit                  ")
    fmt.Println("-----")
    var pilih int

```

```

    fmt.Print("Pilih (1/2): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        mengubahTenant()
    case 2:
        return
    }
}

func mengubahTenant() {
    var pilihTenant, pilihMakanan int
    fmt.Println("Daftar Tenant:")
    for i := 0; i < jumlahTenant; i++ {
        fmt.Printf("%d. %s\n", i+1, tenants[i].NamaTenant)
    }
    fmt.Print("Pilih tenant (1/2/3/...): ")
    fmt.Scan(&pilihTenant)
    if pilihTenant < 1 || pilihTenant > jumlahTenant {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihTenant--

    fmt.Println("Daftar Makanan:")
    for i := 0; i < tenants[pilihTenant].JumlahMakanan; i++ {
        fmt.Printf("%d. %s, Harga: %d, Stok: %d\n", i+1,
tenants[pilihTenant].DataMakanan[i].NamaMakanan,
tenants[pilihTenant].DataMakanan[i].HargaMakanan,
tenants[pilihTenant].DataMakanan[i].Stok)
    }
    fmt.Print("Pilih makanan (1/2/3/...): ")
    fmt.Scan(&pilihMakanan)
    if pilihMakanan < 1 || pilihMakanan > tenants[pilihTenant].JumlahMakanan {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihMakanan--

    fmt.Print("Masukkan nama makanan baru: ")
    fmt.Scan(&tenants[pilihTenant].DataMakanan[pilihMakanan].NamaMakanan)
    fmt.Print("Masukkan harga makanan baru: ")
    fmt.Scan(&tenants[pilihTenant].DataMakanan[pilihMakanan].HargaMakanan)
    fmt.Print("Masukkan stok makanan baru: ")
    fmt.Scan(&tenants[pilihTenant].DataMakanan[pilihMakanan].Stok)
}

```

```

    menu_mengubah_data_tenant()
}

func menu_hapus_data_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Hapus Tenant           ")
    fmt.Println("2. Hapus Data Tenant      ")
    fmt.Println("3. Exit                   ")
    fmt.Println("-----")
    var pilih int
    fmt.Print("Pilih (1/2/3): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        menghapusTenant()
    case 2:
        menghapusDataTenant()
    case 3:
        return
    }
}

func menghapusTenant() {
    var pilihTenant int
    fmt.Println("Daftar Tenant:")
    for i := 0; i < jumlahTenant; i++ {
        fmt.Printf("%d. %s\n", i+1, tenants[i].NamaTenant)
    }
    fmt.Print("Pilih tenant (1/2/3/...): ")
    fmt.Scan(&pilihTenant)
    if pilihTenant < 1 || pilihTenant > jumlahTenant {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihTenant--

    for i := pilihTenant; i < jumlahTenant-1; i++ {
        tenants[i] = tenants[i+1]
    }
    jumlahTenant--
    menu_hapus_data_tenant()
}

```

```

func menghapusDataTenant() {
    var pilihTenant, pilihMakanan int
    fmt.Println("Daftar Tenant:")
    for i := 0; i < jumlahTenant; i++ {
        fmt.Printf("%d. %s\n", i+1, tenants[i].NamaTenant)
    }
    fmt.Print("Pilih tenant (1/2/3/...): ")
    fmt.Scan(&pilihTenant)
    if pilihTenant < 1 || pilihTenant > jumlahTenant {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihTenant--

    fmt.Println("Daftar Makanan:")
    for i := 0; i < tenants[pilihTenant].JumlahMakanan; i++ {
        fmt.Printf("%d. %s, Harga: %d, Stok: %d\n", i+1,
tenants[pilihTenant].DataMakanan[i].NamaMakanan,
tenants[pilihTenant].DataMakanan[i].HargaMakanan,
tenants[pilihTenant].DataMakanan[i].Stok)
    }
    fmt.Print("Pilih makanan (1/2/3/...): ")
    fmt.Scan(&pilihMakanan)
    if pilihMakanan < 1 || pilihMakanan > tenants[pilihTenant].JumlahMakanan {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihMakanan--

    for i := pilihMakanan; i < tenants[pilihTenant].JumlahMakanan-1; i++ {
        tenants[pilihTenant].DataMakanan[i] =
tenants[pilihTenant].DataMakanan[i+1]
    }
    tenants[pilihTenant].JumlahMakanan--
    menu_hapus_data_tenant()
}

func transaksi_data_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Transaksi Tenant       ")
    fmt.Println("2. Total Transaksi Tenant ")
}

```

```

    fmt.Println("3. Cek Uang Admin      ")
    fmt.Println("4. Exit                  ")
    fmt.Println("-----")
    var pilih int
    fmt.Print("Pilih (1/2/3/4): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        transaksiTenant()
    case 2:
        totalTransaksiTenant()
    case 3:
        cekUangAdmin()
    case 4:
        return
    }
}

func transaksiTenant() {
    var pilihTenant, pilihMakanan int
    fmt.Println("Daftar Tenant:")
    for i := 0; i < jumlahTenant; i++ {
        fmt.Printf("%d. %s\n", i+1, tenants[i].NamaTenant)
    }
    fmt.Print("Pilih tenant (1/2/3/...): ")
    fmt.Scan(&pilihTenant)
    if pilihTenant < 1 || pilihTenant > jumlahTenant {
        fmt.Println("Pilihan tidak valid.")
        return
    }
    pilihTenant--

    fmt.Println("Daftar Makanan:")
    for i := 0; i < tenants[pilihTenant].JumlahMakanan; i++ {
        fmt.Printf("%d. %s, Harga: %d, Stok: %d\n", i+1,
tenants[pilihTenant].DataMakanan[i].NamaMakanan,
tenants[pilihTenant].DataMakanan[i].HargaMakanan,
tenants[pilihTenant].DataMakanan[i].Stok)
    }
    fmt.Print("Pilih makanan (1/2/3/...): ")
    fmt.Scan(&pilihMakanan)
    if pilihMakanan < 1 || pilihMakanan > tenants[pilihTenant].JumlahMakanan {
        fmt.Println("Pilihan tidak valid.")
        return
    }
}

```



```

    pilihMakanan--

    if tenants[pilihTenant].DataMakanan[pilihMakanan].Stok == 0 {
        fmt.Println("Stok habis.")
        return
    }
    tenants[pilihTenant].DataMakanan[pilihMakanan].Stok--
    tenants[pilihTenant].TotalTransaksi +=
tenants[pilihTenant].DataMakanan[pilihMakanan].HargaMakanan
    fmt.Println("Transaksi berhasil.")
    transaksi_data_tenant()
}

func totalTranskasiTenant() {
    var totalTransaksi int
    for i := 0; i < jumlahTenant; i++ {
        totalTransaksi += tenants[i].TotalTransaksi
    }
    totalAdmin = totalTransaksi / 4
    fmt.Printf("Total Transaksi Tenant: %d\n", totalTransaksi)
    transaksi_data_tenant()
}

func cekUangAdmin() {
    fmt.Printf("Total Uang Admin (Transaksi): %d\n", totalAdmin)
    fmt.Printf("Total Uang Admin (Sewa): %d\n", totalSewaAdmin)
    transaksi_data_tenant()
}

func tampilkan_data_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Tampilkan Data Tenant ")
    fmt.Println("2. Exit                    ")
    fmt.Println("-----")
    var pilih int
    fmt.Print("Pilih (1/2): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        tampilkanTenant()
    case 2:
        return
    }
}

```

```

    }
}

func tampilkanTenant() {
    fmt.Println("Daftar Tenant:")
    for i := 0; i < jumlahTenant; i++ {
        fmt.Printf("%d. %s\n", i+1, tenants[i].NamaTenant)
        for j := 0; j < tenants[i].JumlahMakanan; j++ {
            fmt.Printf("    %d. %s, Harga: %d, Stok: %d\n", j+1,
tenants[i].DataMakanan[j].NamaMakanan, tenants[i].DataMakanan[j].HargaMakanan,
tenants[i].DataMakanan[j].Stok)
        }
    }
    tampilkan_data_tenant()
}

func urutkan_data_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Urutkan Data Tenant    ")
    fmt.Println("2. Exit                    ")
    fmt.Println("-----")
    var pilih int
    fmt.Print("Pilih (1/2): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        urutanTenant()
    case 2:
        return
    }
}

// << Insertion sort >>
func urutanTenant() {
    for i := 1; i < jumlahTenant; i++ {
        key := tenants[i]
        j := i - 1
        for j >= 0 && tenants[j].TotalTransaksi < key.TotalTransaksi {
            tenants[j+1] = tenants[j]
            j--
        }
        tenants[j+1] = key
    }
}

```

```

    }
    fmt.Println("5 Tenant dengan total transaksi terbanyak:")
    for i := 0; i < jumlahTenant && i < 5; i++ {
        fmt.Printf("%d. %s, Total Transaksi: %d\n", i+1, tenants[i].NamaTenant,
tenants[i].TotalTransaksi)
    }
    urutkan_data_tenant()
}

```

```

func cari_tenant() {
    clear_screen()
    fmt.Println("-----")
    fmt.Println("          M E N U          ")
    fmt.Println("-----")
    fmt.Println("1. Cari Tenant            ")
    fmt.Println("2. Exit                  ")
    fmt.Println("-----")
    var pilih int
    fmt.Print("Pilih (1/2): ")
    fmt.Scan(&pilih)
    switch pilih {
    case 1:
        cariTenantByName()
    case 2:
        return
    }
}

```

```

// << Binary Search >>
func binarySearchTenant(name string) int {
    left, right := 0, jumlahTenant-1
    for left <= right {
        mid := (left + right) / 2
        if tenants[mid].NamaTenant == name {
            return mid
        } else if tenants[mid].NamaTenant < name {
            left = mid + 1
        } else {
            right = mid - 1
        }
    }
    return -1
}

```

```

func cariTenantByName() {

```

```

var name string
fmt.Print("Masukkan nama tenant yang dicari: ")
fmt.Scan(&name)
index := binarySearchTenant(name)
if index != -1 {
    fmt.Printf("Tenant ditemukan: %s\n", tenants[index].NamaTenant)
    for j := 0; j < tenants[index].JumlahMakanan; j++ {
        fmt.Printf("    %d. %s, Harga: %d, Stok: %d\n", j+1,
tenants[index].DataMakanan[j].NamaMakanan,
tenants[index].DataMakanan[j].HargaMakanan, tenants[index].DataMakanan[j].Stok)
    }
} else {
    fmt.Println("Tenant tidak ditemukan.")
}
cari_tenant()
}

// << Selection sort >> untuk mengurutkan tenant berdasarkan nama
func selectionSortTenantsByName() {
    for i := 0; i < jumlahTenant-1; i++ {
        minIndex := i
        for j := i + 1; j < jumlahTenant; j++ {
            if tenants[j].NamaTenant < tenants[minIndex].NamaTenant {
                minIndex = j
            }
        }
        tenants[i], tenants[minIndex] = tenants[minIndex], tenants[i]
    }
    fmt.Println("Data tenant berhasil diurutkan berdasarkan nama.")
    urutkan_data_tenant()
}

```