

**Laporan**  
**Tugas Besar Mata Kuliah Algoritma Pemrograman**  
**Aplikasi Manajemen Stok Bahan Makanan untuk Rumah Tangga**



**Kelompok 12**

**Anggota:**

I Putu Dirga Bramesta Wijana(Ketua) (NIM : 103032400132)

Ahmad Zaky Suhail (NIM : 103032430032)

Zaguna Tora Daivra (NIM: 103032400066)

## PENDAHULUAN

### A. Deskripsi Masalah

Banyak rumah tangga kesulitan mengelola stok bahan makanan sehingga terjadi pemborosan akibat bahan makanan kedaluwarsa atau kekurangan saat dibutuhkan. Aplikasi ini dirancang untuk mencatat, memantau, dan mengelola stok bahan makanan secara efisien.

Dalam kehidupan rumah tangga, pengelolaan bahan makanan merupakan hal penting agar kebutuhan sehari-hari dapat terpenuhi secara efisien dan tidak terjadi pemborosan. Namun, sering kali anggota rumah tangga lupa mencatat stok bahan makanan yang tersedia, lupa tanggal kadaluarsa, atau membeli barang yang sebenarnya masih ada. Hal ini dapat menyebabkan bahan makanan terbuang sia-sia atau kebutuhan memasak menjadi terganggu karena stok habis tanpa disadari.

Oleh karena itu, diperlukan sebuah aplikasi sederhana yang dapat membantu pengguna rumah tangga untuk mencatat, mengelola, dan memantau stok bahan makanan mereka. Aplikasi ini bertujuan untuk memudahkan pengguna dalam:

- Menambahkan stok bahan makanan,
- Menghapus bahan yang sudah tidak digunakan,
- Mengganti jumlah bahan makanan
- Mengurutkan bahan makanan berdasarkan nama dan jumlah stok yang tertera.

Dengan adanya aplikasi ini, diharapkan pengguna bisa lebih teratur dalam mengelola kebutuhan dapur serta mengurangi pemborosan akibat bahan makanan yang terlupa atau kadaluwarsa.

### B. Pembagian Tugas

- Ahmad Zaky Suhail: Bertanggung jawab dalam menyusun laporan akhir proyek serta mengembangkan fungsi utama (main function) yang mengatur alur logika program secara keseluruhan. Fungsi utama ini menjadi pusat koordinasi dari seluruh fungsi lain yang dibuat oleh anggota tim lainnya.
- Zaguna Tora Daivra: Mempunyai tugas untuk merancang dan mengimplementasikan struktur data yang digunakan dalam program, seperti tipe bentukan (struct). Selain itu, juga bertugas membuat fungsi sorting untuk mengurutkan data, serta fungsi tampil untuk menampilkan data yang telah dimasukkan.
- I Putu Dirga Bramesta Wijana : Bertugas mengembangkan beberapa fitur tambahan yaitu fungsi tambah, fungsi kurang (untuk mengurangi atau menghapus stok), dan fungsi cari (untuk mencari data tertentu). Selain itu, juga bertanggung jawab dalam proses ekstraksi file bila dibutuhkan (seperti membaca atau menyimpan data dari/ke file), serta melakukan pengecekan output untuk memastikan program berjalan sesuai dengan yang diharapkan.

# Laporan Proyek: Aplikasi Manajemen Stok Bahan Makanan Sederhana

## 1. Deskripsi Singkat Proyek

Aplikasi ini adalah program command-line sederhana yang dibuat dengan Go untuk mengelola stok bahan makanan. Pengguna dapat menambah, melihat, mengubah jumlah, dan menghapus bahan makanan. Aplikasi juga menyediakan fitur untuk mengurutkan daftar bahan makanan berdasarkan nama atau jumlah, serta mencari bahan makanan tertentu.

## 2. Struktur Data Utama

- **BahanMakanan (struct):**

```
type BahanMakanan struct {  
    Nama      string  
    Jumlah    int  
    Kedaluwarsa time.Time  
}
```

- Nama (string): Nama dari bahan makanan.
- Jumlah (int): Kuantitas bahan makanan yang tersedia.
- Kedaluwarsa (time.Time): Tanggal kedaluwarsa bahan makanan.

- **ArrayBahan (struct):**

```
type ArrayBahan struct {  
    Data [NMAX]BahanMakanan  
    N     int  
}
```

- Data ([NMAX]BahanMakanan): Array untuk menyimpan daftar BahanMakanan.  
NMAX adalah konstanta yang menentukan kapasitas maksimal (100).
- N (int): Jumlah bahan makanan yang saat ini tersimpan dalam array.

## 3. Fungsionalitas Utama

- **Menampilkan Stok (tampilkanStok):**

```
// Menampilkan semua stok bahan
func tampilkanStok(A ArrayBahan) {
    var i int

    fmt.Println("Stok Bahan Makanan:")
    for i = 0; i < A.N; i++ {
        fmt.Printf("%d. %s - %d (Kedaluwarsa: %s)\n", i+1, A.Data[i].Nama, A.Data[i].Jumlah, A.Data[i].Kedaluwarsa.Format("2006-01-02"))
    }
}
```

Mencetak semua bahan makanan yang tersimpan beserta nama, jumlah, dan tanggal kedaluwarsanya.

- **Tambah Bahan Makanan (tambahBahan):**

```
// Tambah bahan
func tambahBahan(A *ArrayBahan, nama string, jumlah int, kadaluwarsa string) {
    var tgl time.Time
    var err error

    if A.N >= NMAX {
        fmt.Println("Stok penuh!")
        return
    }
    tgl, err = time.Parse("2006-01-02", kadaluwarsa)
    if err != nil {
        fmt.Println("Format tanggal salah.")
        return
    }
    A.Data[A.N] = BahanMakanan{nama, jumlah, tgl}
    A.N++
}
```

Memungkinkan pengguna memasukkan nama, jumlah, dan tanggal kedaluwarsa (format YYYY-MM-DD) untuk bahan makanan baru. Data akan ditambahkan ke dalam array stok jika kapasitas belum penuh.

- **Ubah Jumlah Bahan (ubahBahan):**

```
// Ubah data (sequential search)
func ubahBahan(A *ArrayBahan, nama string, jumlahBaru int) {
    var idx int

    idx = cariSequential(*A, nama)
    if idx != -1 {
        A.Data[idx].Jumlah = jumlahBaru
        fmt.Println("Data berhasil diubah.")
    } else {
        fmt.Println("Bahan tidak ditemukan.")
    }
}
```

Mengubah jumlah bahan makanan yang sudah ada. Pengguna memasukkan nama bahan dan jumlah baru. Pencarian bahan menggunakan sequential search (case-insensitive).

- **Hapus Bahan Makanan (hapusBahan):**

```
// Hapus data (binary search setelah diurutkan)
func hapusBahan(A *ArrayBahan, nama string) {
    var i, idx int

    insertionSortNama(A, true)
    idx = binarySearch(*A, nama)
    if idx != -1 {
        for i = idx; i < A.N-1; i++ {
            A.Data[i] = A.Data[i+1]
        }
        A.N--
        fmt.Println("Data berhasil dihapus.")
    } else {
        fmt.Println("Data tidak ditemukan.")
    }
}
```

Menghapus bahan makanan dari daftar. Program akan terlebih dahulu mengurutkan data berdasarkan nama (menggunakan Insertion Sort), lalu mencari bahan menggunakan binary search (case-insensitive), dan kemudian menghapusnya.

- **Sortir Stok:**

- **selectionSortJumlah:**

```
// Selection Sort berdasarkan jumlah
func selectionSortJumlah(A *ArrayBahan, ascending bool) {
    var i, j, idx int

    for i = 0; i < A.N-1; i++ {
        idx = i
        for j = i + 1; j < A.N; j++ {
            if (ascending && A.Data[j].Jumlah < A.Data[idx].Jumlah) || (!ascending && A.Data[j].Jumlah > A.Data[idx].Jumlah) {
                idx = j
            }
        }
        A.Data[i], A.Data[idx] = A.Data[idx], A.Data[i]
    }
}
```

Mengurutkan stok berdasarkan jumlah (bisa ascending atau descending).

- **insertionSortNama:**

```
// Insertion Sort berdasarkan nama
func insertionSortNama(A *ArrayBahan, ascending bool) {
    var i, j int
    var key BahanMakanan

    for i = 1; i < A.N; i++ {
        key = A.Data[i]
        j = i - 1
        for j >= 0 && ((ascending && A.Data[j].Nama > key.Nama) || (!ascending && A.Data[j].Nama < key.Nama)) {
            A.Data[j+1] = A.Data[j]
            j--
        }
        A.Data[j+1] = key
    }
}
```

Mengurutkan stok berdasarkan nama (bisa ascending atau descending).

- **Pencarian (Internal):**

- **cariSequential:**

```
// Sequential Search
func cariSequential(A ArrayBahan, nama string) int {
    var i int

    for i = 0; i < A.N; i++ {
        if samaTanpaCase(A.Data[i].Nama, nama) {
            return i
        }
    }
    return -1
}
```

Pencarian sekuensial yang digunakan untuk mengubah data.

- **binarySearch:**

```
// Binary Search (harus terurut berdasarkan nama ascending)
func binarySearch(A ArrayBahan, nama string) int {
    var low, high, mid int

    low, high = 0, A.N-1
    for low <= high {
        mid = (low + high) / 2
        if samaTanpaCase(A.Data[mid].Nama, nama) {
            return mid
        } else if A.Data[mid].Nama < nama {
            low = mid + 1
        } else {
            high = mid - 1
        }
    }
    return -1
}
```

Pencarian biner yang digunakan setelah data diurutkan untuk menghapus data.

- **samaTanpaCase:**

```
// Fungsi pembandingan string tanpa strings.EqualFold
func samaTanpaCase(a, b string) bool {
    if len(a) != len(b) {
        return false
    }
    for i := 0; i < len(a); i++ {
        pa, pb := a[i], b[i]
        if pa >= 'A' && pa <= 'Z' {
            pa += 32
        }
        if pb >= 'A' && pb <= 'Z' {
            pb += 32
        }
        if pa != pb {
            return false
        }
    }
    return true
}
```

Fungsi kustom untuk perbandingan string tanpa membedakan huruf besar/kecil, digunakan dalam kedua fungsi pencarian.

## • Program Utama

```
164 // Main Program
165 func main() {
166     var stok ArrayBahan
167
168     // Data awal
169     tambahBahan(&stok, "Beras", 10, "2025-12-01")
170     tambahBahan(&stok, "Gula", 5, "2025-11-01")
171     tambahBahan(&stok, "Minyak", 3, "2025-10-15")
172     tambahBahan(&stok, "Sayur", 8, "2025-06-15")
173
174     var pilihan int
175     for {
176         fmt.Println("\nMenu:")
177         fmt.Println("1. Tampilkan Stok")
178         fmt.Println("2. Tambah Bahan Makanan")
179         fmt.Println("3. Ubah Jumlah ")
180         fmt.Println("4. Hapus Bahan ")
181         fmt.Println("5. Sort Jumlah ")
182         fmt.Println("6. Sort Nama ")
183         fmt.Println("7. Keluar")
184         fmt.Print("Pilihan: ")
185         fmt.Scan(&pilihan)
186
187         switch pilihan {
188             case 1:
189                 tampilkanStok(stok)
190             case 2:
191                 var nama, tgl string
192                 var jumlah int
193                 fmt.Print("Nama: ")
194                 fmt.Scan(&nama)
195                 fmt.Print("Jumlah: ")
196                 fmt.Scan(&jumlah)
197                 fmt.Print("Tanggal Kadaluwarsa (YYYY-MM-DD): ")
198                 fmt.Scan(&tgl)
199                 tambahBahan(&stok, nama, jumlah, tgl)
200             case 3:
201                 var nama string
202                 var jumlah int
203                 fmt.Print("Nama bahan: ")
204                 fmt.Scan(&nama)
205                 fmt.Print("Jumlah baru: ")
206                 fmt.Scan(&jumlah)
207                 ubahBahan(&stok, nama, jumlah)
208
209             case 4:
210                 var nama string
211                 fmt.Print("Nama bahan yang dihapus: ")
212                 fmt.Scan(&nama)
213                 hapusBahan(&stok, nama)
214             case 5:
215                 var asc int
216                 fmt.Print("1. Ascending, 2. Descending: ")
217                 fmt.Scan(&asc)
218                 selectionSortJumlah(&stok, asc == 1)
219                 fmt.Println("Stok diurutkan berdasarkan jumlah.")
220             case 6:
221                 var asc int
222                 fmt.Print("1. Ascending, 2. Descending: ")
223                 fmt.Scan(&asc)
224                 insertionSortNama(&stok, asc == 1)
225                 fmt.Println("Stok diurutkan berdasarkan nama.")
226             case 7:
227                 fmt.Println("Terima kasih.")
228                 return
229         }
230     }
```



## I. Tipe Data

### a. Tipe Data Bawaan (Primitive)

int: digunakan untuk menyimpan jumlah bahan dan pilihan menu.

string: menyimpan nama bahan dan tanggal.

bool: digunakan dalam sorting untuk menentukan ascending atau descending.

### b. Tipe Data Waktu

time.Time: dari paket time, menyimpan tanggal kedaluwarsa bahan makanan.

### c. Tipe Data Buatan Sendiri (Struct)

go

```
type BahanMakanan struct {  
    Nama      string  
    Jumlah    int  
    Kedaluwarsa time.Time  
}
```

```
type ArrayBahan struct {  
    Data [NMAX]BahanMakanan  
    N     int  
}
```

Digunakan untuk menyimpan dan mengelola daftar stok bahan makanan secara terstruktur.

## II. Perulangan (Looping)

### a. for loop standar

Digunakan di banyak tempat, misalnya:

Menampilkan semua bahan makanan.

Menelusuri array untuk pencarian dan pengurutan.

Menu utama (perulangan tanpa kondisi akhir):

go

```
for {
```

```
// tampilan menu  
}
```

#### **b. for sebagai while-loop**

```
go  
for j >= 0 && condition {  
    // digunakan dalam insertion sort  
}
```

### **III. Percabangan (Branching)**

#### **a. if, else if, dan else**

Digunakan untuk:

Validasi tanggal.

Mencocokkan string.

Menentukan urutan sort ascending/descending.

Menentukan hasil pencarian (ditemukan/tidak ditemukan).

#### **b. switch**

Digunakan dalam menu utama:

```
go  
switch pilihan {  
case 1:  
    // tampilkan stok  
case 2:
```

```
// tambah bahan  
  
...  
  
}
```

#### **IV. Fungsi dan Prosedur**

Kode menggunakan modularisasi dengan berbagai fungsi seperti:

tampilkanStok – menampilkan data.

tambahBahan, ubahBahan, hapusBahan – manajemen data.

samaTanpaCase – membandingkan string tanpa memperhatikan huruf besar/kecil.

cariSequential, binarySearch – pencarian data.

selectionSortJumlah, insertionSortNama – pengurutan data.

#### **V. Algoritma Khusus**

Sequential Search: untuk pencarian linear.

Binary Search: untuk pencarian setelah data diurutkan.

Selection Sort: untuk mengurutkan berdasarkan jumlah.

Insertion Sort: untuk mengurutkan berdasarkan nama.

#### **VI. Konstanta**

go

const NMAX = 100

Digunakan untuk membatasi jumlah maksimum data yang bisa disimpan.

## 1. Tampilkan Stok

```
Microsoft Windows [Version 10.0.22631.5335]
(c) Microsoft Corporation. All rights reserved.

D:\UNIV\Telkom\Tugas\Semester 2\ALPRO\TUBES>go run tugasbesarrrr.go

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: |
```

Menu Awal

## 2. Tambah Bahan Makanan

```
Microsoft Windows [Version 10.0.22631.5335]
(c) Microsoft Corporation. All rights reserved.

D:\UNIV\Telkom\Tugas\Semester 2\ALPRO\TUBES>go run tugasbesarrrr.go

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Minyak - 3 (Kedaluwarsa: 2025-10-15)
4. Sayur - 8 (Kedaluwarsa: 2025-06-15)

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: |
```

Case 1 menampilkan bahan makan

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 2
Nama: Mie
Jumlah: 5
Tanggal Kadaluwarsa (YYYY-MM-DD): 2025-06-01
```

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Minyak - 3 (Kedaluwarsa: 2025-10-15)
4. Sayur - 8 (Kedaluwarsa: 2025-06-15)
5. Mie - 5 (Kedaluwarsa: 2025-06-01)
```

Case 2(berhasil)

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 2
Nama: Ayam
Jumlah: 2
Tanggal Kadaluwarsa (YYYY-MM-DD): 2026-6-1
Format tanggal salah.
```

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Minyak - 3 (Kedaluwarsa: 2025-10-15)
4. Sayur - 8 (Kedaluwarsa: 2025-06-15)
5. Mie - 5 (Kedaluwarsa: 2025-06-01)
```

Case 2(gagal)

### 3. Ubah Jumlah

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 3
Nama bahan: Mie
Jumlah baru: 4
Data berhasil diubah.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Minyak - 3 (Kedaluwarsa: 2025-10-15)
4. Sayur - 8 (Kedaluwarsa: 2025-06-15)
5. Mie - 4 (Kedaluwarsa: 2025-06-01)
```

Case 3(berhasil)

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 3
Nama bahan: Ayam
Jumlah baru: 1
Bahan tidak ditemukan.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Minyak - 3 (Kedaluwarsa: 2025-10-15)
4. Sayur - 8 (Kedaluwarsa: 2025-06-15)
5. Mie - 4 (Kedaluwarsa: 2025-06-01)
```

Case 3(gagal)

## 4. Hapus Bahan

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)
5. Sayur - 8 (Kedaluwarsa: 2025-06-15)

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 4
Nama bahan yang dihapus: Sayur
Data berhasil dihapus.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)
```

Case 4(berhasil)



```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 4
Nama bahan yang dihapus: Ayam
Data tidak ditemukan.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)
```

Case 4(gagal)

## 5. Sortir Berdasarkan Jumlah

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 5
1. Ascending, 2. Descending: 1
Stok diurutkan berdasarkan jumlah.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Minyak - 3 (Kedaluwarsa: 2025-10-15)
2. Mie - 4 (Kedaluwarsa: 2026-06-01)
3. Gula - 5 (Kedaluwarsa: 2025-11-01)
4. Beras - 10 (Kedaluwarsa: 2025-12-01)
```

Case 5(ascending)

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Minyak - 3 (Kedaluwarsa: 2025-10-15)
2. Mie - 4 (Kedaluwarsa: 2026-06-01)
3. Gula - 5 (Kedaluwarsa: 2025-11-01)
4. Beras - 10 (Kedaluwarsa: 2025-12-01)
```

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 5
1. Ascending, 2. Descending: 2
Stok diurutkan berdasarkan jumlah.
```

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)
```

Case 5(descending)

## 6. Sortir Berdasarkan Nama

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 6
1. Ascending, 2. Descending: 1
Stok diurutkan berdasarkan nama.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)
```

Case 6(ascending)

```
Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Beras - 10 (Kedaluwarsa: 2025-12-01)
2. Gula - 5 (Kedaluwarsa: 2025-11-01)
3. Mie - 4 (Kedaluwarsa: 2026-06-01)
4. Minyak - 3 (Kedaluwarsa: 2025-10-15)

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 6
1. Ascending, 2. Descending: 2
Stok diurutkan berdasarkan nama.

Menu:
1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar
Pilihan: 1
Stok Bahan Makanan:
1. Minyak - 3 (Kedaluwarsa: 2025-10-15)
2. Mie - 4 (Kedaluwarsa: 2026-06-01)
3. Gula - 5 (Kedaluwarsa: 2025-11-01)
4. Beras - 10 (Kedaluwarsa: 2025-12-01)
```

Case 6(descending)

## 7. Keluar

Menu:

1. Tampilkan Stok
2. Tambah Bahan Makanan
3. Ubah Jumlah
4. Hapus Bahan
5. Sort Jumlah
6. Sort Nama
7. Keluar

Pilihan: 7

Terima kasih.

D:\UNIV\Telkom\Tugas\Semester 2\ALPRO\TUBES>|