

Struktur Data:

Pada program yang kami buat, kami menggunakan **struktur data linked list** untuk menyimpan daftar seluruh kota.

Setiap node kota berisi nama kota, pointer ke daftar warga (nextWarga), dan pointer ke kota berikutnya (nextKota).

Setiap warga juga disimpan dalam bentuk linked list, di mana setiap node berisi nama warga dan pointer ke warga berikutnya (next).

```
typedef struct DataWarga {
    char namaWarga[MAX_NAMA];
    struct DataWarga *next;
} DataWarga;

typedef struct Kota {
    char NamaKota[MAX_NAMA];
    DataWarga *nextWarga;
    struct Kota *nextKota;
} Kota;
```

Alur Program:

Program akan menampilkan 7 menu, dan setiap menu akan mengeksekusi perintah-perintah yang berbeda.

Menu 1

Jika user memilih menu 1, maka program akan meminta nama kota dari user dan memanggil fungsi EntryData.

Di dalam EntryData, akan dibuat node kota baru menggunakan BuatKotaBaru, lalu menambahkannya ke akhir daftar kota.

Jika daftar kota masih kosong, maka kota baru akan menjadi kota pertama (firstKota).

Menu 2

Jika user memilih menu 2, maka program akan meminta nama kota dan nama warga dari user, lalu memanggil fungsi TambahWargaKeKota.

Fungsi ini akan mencari kota berdasarkan nama yang diinputkan.

Jika kota ditemukan, maka program akan memanggil fungsi TambahWarga untuk membuat node warga baru (alokasi memori) dan menambahkannya ke akhir linked list warga di kota tersebut.

Menu 3

Jika user memilih menu 3, maka program akan memanggil fungsi TampilkanSemua. Fungsi ini akan mencetak semua nama kota yang ada di daftar, serta semua nama warga dari masing-masing kota satu per satu (jika ada).

Menu 4

Jika user memilih menu 4, maka program akan meminta nama kota dari user dan memanggil fungsi TampilkanWarga.

Fungsi ini akan mencari kota berdasarkan nama yang diinputkan, lalu menampilkan semua warga yang terdaftar pada kota tersebut (jika ada).

Menu 5

Jika user memilih menu 5, maka program akan meminta nama kota dan nama warga dari user, lalu memanggil fungsi HapusWargaDariKota.

Di dalam fungsi ini, program akan mencari kota yang sesuai, lalu memanggil HapusWarga untuk mencari nama warga yang sesuai dan menghapus node tersebut dari linked list warga, serta melakukan free untuk membebaskan memori.

Menu 6

Jika user memilih menu 6, maka program akan meminta nama kota dari user dan memanggil fungsi HapusKota.

Fungsi ini akan mencari kota berdasarkan nama yang diinputkan, lalu menghapus semua warga yang ada di kota tersebut dengan memanggil HapusSemuaWarga. Setelah itu, node kota akan dihapus dan pointer dari kota sebelumnya akan disambungkan ke kota berikutnya.

Menu 7

Jika user memilih menu 7, maka program akan keluar dan program akan berhenti.

HASIL SCREENSHOOT

Menu 1

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 1

Masukkan nama kota: bandung

Kota 'bandung' berhasil ditambahkan.

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 3

Kota: bandung

Tidak ada warga

Menu 2

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 2

Masukkan nama kota: bandung

Masukkan nama warga: bayu

Warga 'bayu' berhasil ditambahkan ke kota 'bandung'.

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 3

Kota: bandung

- bayu

Menu 3

```
Menu:
1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar
Pilih menu: 3

Kota: bandung
    - bayu
```

Menu 4

```
Menu:
1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar
Pilih menu: 4

Masukkan nama kota: bandung

Warga di kota bandung:
    - bayu
```

Menu 5

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 5

Masukkan nama kota: bandung

Masukkan nama warga: bayu

Warga 'bayu' berhasil dihapus dari kota 'bandung'.

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 3

Kota: bandung

Tidak ada warga

Menu 6

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 6

Masukkan nama kota yang mau dihapus: bandung

Kota 'bandung' berhasil dihapus beserta semua warganya.

Menu:

1. Tambah Kota
2. Tambah Warga ke Kota
3. Tampilkan Semua Kota dan Warganya
4. Tampilkan Warga dalam Satu Kota
5. Hapus Warga dari Kota
6. Hapus Kota
7. Keluar

Pilih menu: 3

Belum ada kota.