

Penjelasan Alur Program

1. Inisiasi Array

Di awal program, akan dibuat array bertipe struct kota yang berisi 5 elemen. Array ini masih kosong (["" | NULL]), sehingga jika diprint hasilnya:

```
[]  
[]  
[]  
[]  
[]
```

2. Memilih opsi

Program akan menampilkan opsi yang bisa dipilih user

```
===== Operasi Kota =====  
1. Insert Kota  
2. Delete Kota  
3. Print All  
===== Operasi Warga =====  
4. Insert Warga  
5. Delete Warga  
6. Print Warga  
===== Exit =====  
7. Exit  
Pilih(1-7):
```

3. Opsi 1 : Insert Kota

Jika memilih opsi ini, user akan diminta memasukkan nama kota. Misal, Bandung

```
Pilih(1-7): 1  
  
Masukkan nama kota: Bandung  
Tekkan enter untuk melanjutkan...
```

Maka array akan berisi:

[Bandung] -> NULL

[] -> NULL

[] -> NULL

[] -> NULL

[] -> NULL

4. Opsi 2 : Delete Kota

Pada opsi ini, user dapat memilih kota untuk dihapus. Jika menghapus kota, maka warga otomatis terhapus seluruhnya di kota tersebut.

Misal terdapat data:

```
[Bandung][Arief]
[Garut][Maulana]->[Tsinan]
[Karawang][Fanezi]
[Jakarta]
[Bekasi]
```

Lalu saya memilih opsi 2, dan mendelete kota Garut. Maka:

```
Pilih(1-7): 2
```

```
Masukkan nama kota untuk dihapus: Garut
Tekkan enter untuk melanjutkan...
```

Hasil Arraynya:

```
[Bandung][Arief]
[]
[Karawang][Fanezi]
[Jakarta]
[Bekasi]
```

Jika kota yang di-inputkan tidak ada di list, maka:

```
Pilih(1-7): 2
```

```
Masukkan nama kota untuk dihapus: Tasikmalaya
Kota Tasikmalaya tidak ada di list
Tekkan enter untuk melanjutkan...
```

5. Opsi 3 : Print All

Jika memilih opsi ini, program akan menampilkan semua kota beserta warganya:

```
Pilih(1-7): 3
```

```
[Bandung][Arief]
```

```
[]
```

```
[Karawang][Fanezi]
```

```
[Jakarta]
```

```
[Bekasi]
```

```
Tekkan enter untuk melanjutkan...
```

6. Opsi 4: Insert Warga

Opsi ini akan meminta nama kota dan nama warga untuk dimasukkan ke list.

Misal terdapat kota bandung:

```
[Bandung]
```

```
[]
```

```
[]
```

```
[]
```

```
[]
```

Lalu, saya ingin memasukkan warga bernama "Maulana" ke kota Bandung tsb.

Maka:

```
Pilih(1-7): 4
```

```
Masukkan nama kota: Bandung
```

```
Masukkan nama warga: Maulana
```

```
Tekkan enter untuk melanjutkan...
```

Hasilnya:

```
[Bandung][Maulana]
```

```
[]
```

```
[]
```

```
[]
```

```
[]
```

Jika saya mencoba memasukkan "Maulana" ke kota yang tidak lain yang tidak ada, maka:

```
Pilih(1-7): 4

Masukkan nama kota: Tasikmalaya
Masukkan nama warga: Maulana
Kota Tasikmalaya tidak ada di list
```

7. Opsi 5 : Delete warga

Jika memilih opsi ini, warga di kota tertentu dapat dihapus. Misal saya ingin menghapus “Maulana” di kota Bandung tadi, maka:

```
Pilih(1-7): 5

Masukkan nama kota asal: Bandung
Masukkan nama warga: Maulana

Warga Dihapus: Maulana
Tekkan enter untuk melanjutkan...
```

Hasilnya:

```
[Bandung]
[]
[]
[]
[]
```

Jika saya mencoba menghapus “Maulana” lagi, padahal sudah tidak ada di list, maka yang terjadi:

```
Pilih(1-7): 5

Masukkan nama kota asal: Bandung
Masukkan nama warga: Maulana
Maulana tidak ditemukan
```

8. Opsi 6 : Print Warga

Opsi ini membuat program melakukan print warga di kota tertentu saja. Misal saya memiliki data:

```
[Bandung][Maulana]->[Aried]->[Andi]  
[garut][Tsinan]->[Akmal]  
[Bekasi][Zidan]  
[Jakarta]  
[Depok][Wafi]
```

Lalu, saya ingin menampilkan warga di Kota Bandung saja, maka:

```
7. EXIT  
Pilih(1-7): 6  
  
Masukkan nama kota: Bandung  
  
[Maulana]->[Aried]->[Andi]  
Tekkan enter untuk melanjutkan...
```

Jika kotanya tidak ada di list, maka:

```
Pilih(1-7): 6  
  
Masukkan nama kota: Tangerang  
  
kota Tangerang tidak ada di list.
```

9. Opsi 7 : Exit

Opsi ini berarti keluar dari program