

TP MOD 10 STRUKTUR DATA

1. Flight.h

```

main.cpp X flight.h X flight.cpp X
1  #ifndef FLIGHT_H_INCLUDED
2  #define FLIGHT_H_INCLUDED
3
4  #include <iostream>
5  using namespace std;
6
7  struct jadwalPenerbangan{
8      string Kode;
9      string Jenis;
10     string Tanggal;
11     string Waktu;
12     string Asal;
13     string Tujuan;
14     int Kapasitas;
15 };
16
17 typedef struct elemenJadwal *adr_jadwalP;
18
19 typedef jadwalPenerbangan infotype;
20
21 struct elemenJadwal{
22     infotype info;
23     adr_jadwalP next;
24 };
25
26 struct ListJadwal {
27     adr_jadwalP First;
28 };
29
30 void createListJadwal_103032330095(ListJadwal &L);
31
32 adr_jadwalP createElemenJadwal_103032330095(infotype X);
33
34 void InsertLastJ_103032330095(ListJadwal &L, adr_jadwalP P);
35
36 void ShowJadwal__103032330095(ListJadwal L);
37
38 void deleteFirstJ__103032330095(ListJadwal &L, adr_jadwalP &P);
39
40 adr_jadwalP SearchJ__103032330095(ListJadwal L, string dari, string ke, string tanggal);
41
42
43 #endif // FLIGHT_H_INCLUDED

```

2. Flight.cpp

```

main.cpp x flight.h x flight.cpp x
1      #include "flight.h"
2
3      void createListJadwal_103032330095(ListJadwal &L){
4          L.First = NULL;
5      }
6
7      adr_jadwalP createElemenJadwal_103032330095(infotype X){
8          adr_jadwalP P = new elemenJadwal;
9          P->info = X;
10         P->next = NULL;
11         return P;
12     }
13
14     void InsertLastJ_103032330095(ListJadwal &L, adr_jadwalP P){
15         adr_jadwalP Q;
16         if (L.First == NULL){
17             L.First = P;
18         }else{
19             Q = L.First;
20             while (Q->next != NULL){
21                 Q = Q->next;
22             }
23             Q->next = P;
24         }
25     }
26
27     void ShowJadwal_103032330095(ListJadwal L){
28         adr_jadwalP P;
29         if (L.First == NULL){
30             cout << "List Kosong!" << endl;
31         }else{
32             P = L.First;
33             while (P!= NULL){
34                 cout << "Kode Penerbangan : " << P->info.Kode << " || " ;
35                 cout << "Jenis Penerbangan : " << P->info.Jenis << " || ";
36                 cout << "Tanggal Penerbangan : " << P->info.Tanggal << " || ";
37                 cout << "Waktu Penerbangan : " << P->info.Waktu << " || ";
38                 cout << "Asal Penerbangan : " << P->info.Asal << " || ";
39                 cout << "Tujuan Penerbangan : " << P->info.Tujuan << " || ";
40                 cout << "Kapasitas Penerbangan : " << P->info.Kapasitas << endl;
41                 cout << endl;
42                 P = P->next;
43             }
44         }
45     }

```

```

main.cpp x flight.h x flight.cpp x
46
47 void deleteFirstJ_103032330095(ListJadwal &L, adr_jadwalP &P){
48     if (L.First == NULL){
49         P = NULL;
50     }else if (L.First->next == NULL){
51         P = L.First;
52         L.First = NULL;
53     }else{
54         P = L.First;
55         L.First = P->next;
56         P->next = NULL;
57     }
58 }
59
60 adr_jadwalP SearchJ_103032330095(ListJadwal L, string dari, string ke, string tanggal){
61     adr_jadwalP P;
62     bool ketemu = false;
63     P = L.First;
64     while (P != NULL && !ketemu){
65         ketemu = P->info.Asal == dari && P->info.Tujuan == ke && P->info.Tanggal == tanggal;
66         if (!ketemu){
67             P = P->next;
68         }
69     }
70     return P;
71 }

```

3. Main.cpp

```

main.cpp x flight.h x flight.cpp x
1 #include "flight.h"
2
3 int main()
4 {
5     ListJadwal L;
6     infotype X;
7     adr_jadwalP P;
8     int n, i;
9     string dari, ke, tanggal;
10
11     createListJadwal_103032330095(L);
12     cout << "N : ";
13     cin >> n;
14
15     for (i = 1; i <= n; i++){
16         cout << "Kode Penerbangan : ";
17         cin >> X.Kode;
18         cout << "Jenis Penerbangan : ";
19         cin >> X.Jenis;
20         cout << "Tanggal Penerbangan : ";
21         cin >> X.Tanggal;
22         cout << "Waktu Penerbangan : ";
23         cin >> X.Waktu;
24         cout << "Asal Penerbangan : ";
25         cin >> X.Asal;
26         cout << "Tujuan Penerbangan : ";
27         cin >> X.Tujuan;
28         cout << "Kapasitas Penerbangan : ";
29         cin >> X.Kapasitas;
30         cout << endl;

```

```

main.cpp x flight.h x flight.cpp x
31     P = createElemenJadwal_103032330095(X);
32     InsertLastJ_103032330095(L, P);
33 }
34
35     cout << "Data List Jadwal awal : " << endl;
36     ShowJadwal__103032330095(L);
37
38     cout << "Data setelah dilakukan penghapusan : " << endl;
39     deleteFirstJ_103032330095(L, P);
40     ShowJadwal__103032330095(L);
41
42     cout << "Cari jadwal penerbangan : " << endl;
43     cout << "Dari : ";
44     cin >> dari;
45     cout << "Ke : ";
46     cin >> ke;
47     cout << "Tanggal : ";
48     cin >> tanggal;
49
50     P = SearchJ__103032330095(L, dari, ke, tanggal);
51
52     if (P != NULL){
53         cout << endl << "Jadwal penerbangan ditemukan!" << endl;
54         cout << "Kode Penerbangan : " << P->info.Kode << " || " ;
55         cout << "Jenis Penerbangan : " << P->info.Jenis << " || " ;
56         cout << "Tanggal Penerbangan : " << P->info.Tanggal << " || " ;
57         cout << "Waktu Penerbangan : " << P->info.Waktu << " || " ;
58         cout << "Asal Penerbangan : " << P->info.Asal << " || " ;
59         cout << "Tujuan Penerbangan : " << P->info.Tujuan << " || " ;
60         cout << "Kapasitas Penerbangan : " << P->info.Kapasitas << endl;
61         cout << endl;
62     } else {
63         cout << endl << "Jadwal penerbangan tidak ditemukan!" << endl;
64     }
65     return 0;
66 }
67

```

4. Output

```
"D:\KuliahGena\SMT 3\Strukt x + v
N : 2
Kode Penerbangan : 01
Jenis Penerbangan : Domestik
Tanggal Penerbangan : 11_November_2024
Waktu Penerbangan : 21:00_WITA
Asal Penerbangan : Denpasar
Tujuan Penerbangan : Bandung
Kapasitas Penerbangan : 170

Kode Penerbangan : 02
Jenis Penerbangan : Domestik
Tanggal Penerbangan : 09_Desember_2022
Waktu Penerbangan : 07:00_WIB
Asal Penerbangan : Surabaya
Tujuan Penerbangan : Malang
Kapasitas Penerbangan : 240

Data List Jadwal awal :
Kode Penerbangan : 01 || Jenis Penerbangan : Domestik || Tanggal Penerbangan : 11_November_2024
|| Waktu Penerbangan : 21:00_WITA || Asal Penerbangan : Denpasar || Tujuan Penerbangan : Bandu
ng || Kapasitas Penerbangan : 170

Kode Penerbangan : 02 || Jenis Penerbangan : Domestik || Tanggal Penerbangan : 09_Desember_2022
|| Waktu Penerbangan : 07:00_WIB || Asal Penerbangan : Surabaya || Tujuan Penerbangan : Malang
|| Kapasitas Penerbangan : 240

Data setelah dilakukan penghapusan :
Kode Penerbangan : 02 || Jenis Penerbangan : Domestik || Tanggal Penerbangan : 09_Desember_2022
|| Waktu Penerbangan : 07:00_WIB || Asal Penerbangan : Surabaya || Tujuan Penerbangan : Malang
|| Kapasitas Penerbangan : 240

Cari jadwal penerbangan :
Dari : Surabaya
Ke : Malang
Tanggal : 09_Desember_2022

Jadwal penerbangan ditemukan!
Kode Penerbangan : 02 || Jenis Penerbangan : Domestik || Tanggal Penerbangan : 09_Desember_2022
|| Waktu Penerbangan : 07:00_WIB || Asal Penerbangan : Surabaya || Tujuan Penerbangan : Malang
|| Kapasitas Penerbangan : 240

Process returned 0 (0x0) execution time : 26.215 s
Press any key to continue.
|
```