

```

1  #ifndef BRAIN_H_INCLUDED
2  #define BRAIN_H_INCLUDED
3  #include <iostream>
4  #include <stdlib.h>
5  #include <stdio.h>
6  #define info(P) (P)->info
7  #define next(P) (P)->next
8  #define first(L) ((L).first)
9  #define col(P) (P)->col
10 using namespace std;
11
12 //ADT
13 struct infopasien{
14     string nama;
15     string sakit;
16 };
17 struct infoobat{
18     string obat;
19     string jenis;
20 };
21 typedef struct eklm *address_kolom;
22 typedef struct eklm{
23     infoobat info;
24     address_kolom next;
25 }ekolom;
26 typedef struct ebrs *address_baris;
27 typedef struct ebrs{
28     infopasien info;
29     ekolom *col;
30     address_baris next;
31 }ebaris;
32 struct List{
33     address_baris first;
34 };
35 struct listobat{
36     address_kolom first;
37 };
38
39 address_baris AdrBaris(infopasien x);
40 address_kolom AdrKolom(infoobat x);
41 void createlist(List &L);
42 void createlistobat(listobat &L);
43 void DaftarObat(listobat &L);
44 void printObat(listobat L, string x);
45 void printPenyakit(listobat L);
46 address_kolom findobat(listobat L, string x);
47 address_baris findpasien(List L, string x);
48 address_kolom findJenisObat(listobat L, string x);
49
50 void addFirstBaris(infopasien pasien, List &L);
51 void addFirstKolom(infoobat obat, List &L);
52 void printPasienObat(List L);
53
54 void addTKolom(string pasien, address_kolom Q, List &L);
55 bool sudahPunya(List L, string obat, address_baris P);
56
57 //Procedure DELETE
58 void CekPosisiDel(List &L, address_baris P);
59 void DelFirst(List &L);
60 void DelLast(List &L);
61 void DelTengah(List &L, string x);
62
63 void ShowSatuPasien(List L, address_baris P);
64 void coutObatBanyak(List L, listobat O);
65 void printObat3(List L);
66 #endif // BRAIN_H_INCLUDED
67

```