
Länkad lista med huvud och rekursion

I denna kod innehåller den länkade listan ett *huvud*. I och med det behöver inte insättning och borttagning av första posten i listan inte särbehandlas. Att skriva rekursiva rutiner blir enkelt.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 struct perstyp{
5     char namn[10];
6     struct perstyp *pekare;
7 };
8
9 typedef struct perstyp person;
10
11 person *initlist(void){
12     return (person *)malloc(sizeof(person));
13 }
14
15 person laesin(char filnamn[], person *pers){
16     FILE *infil;
17     int antal,i;
18     person *ny;
19
20     infil=fopen(filnamn,"rt");
21     fscanf(infil,"%d",&antal);
22
23     for(i=1;i<=antal;i++){
24         ny=(person *)malloc(sizeof(person));
25         fscanf(infil,"%s",ny->namn);
26         pers->pekare=ny;
27         pers=ny;
28     }
29     pers->pekare=NULL;
30     fclose(infil);
31 }
32
33 void skrivut(person *pers){
34     if(pers->pekare!=NULL){
35         printf("%s\n",pers->pekare->namn);
36         skrivut(pers->pekare);
37     }
38 }
39
40 void tabort(person *pers,char namn[]){
41     person *tmp;
42     if(pers->pekare!=NULL)
43         if(strcmp(namn,pers->pekare->namn)==0){
44             tmp=pers->pekare;
45             pers->pekare=pers->pekare->pekare;
46             free(tmp);
47         }
48     else
49         tabort(pers->pekare,namn);
50 }
```

```
1 void freelist(person *pers){
2     person *tmp;
3     if(pers!=NULL){
4         tmp=pers;
5         free(pers);
6         freelist(pers->pekare);
7     }
8 }
9
10 void main(void){
11     person *start;
12     start=initlist();
13     laesin("namn.txt",start);
14     skrivut(start);
15     tabort(start,"Birgitta");
16     tabort(start,"Anna");
17     printf("-----\n");
18     skrivut(start);
19     freelist(start);
20 }
```