

```
//Menu alapjan mukodo program veszi az elemeket az adatok.txt-bol, neha pop() muveletet hajta  
//vegre, neha push() muveletet. Amit pop()-al kiemelt azt irja kepernyore es legyen listazas  
//menupont is.
```

```
#include <iostream>  
#include <cstdlib>  
#include <fstream>
```

```
using namespace std;  
ifstream f("adatok.txt");
```

```
typedef struct elem  
{  
    int adat;  
    elem *kov;  
};
```

```
void init(elem *&e, int ertek)
```

```
{  
    e=new elem;  
    e->adat=ertek;  
    e->kov=NULL;  
}
```

```
void hozzaad(elem *e, int ertek) //e: listafej
```

```
{  
    elem *p, *q;  
    p=new elem;  
    p->adat=ertek;  
    q=e;  
    while (q->kov!=NULL) q=q->kov;  
    q->kov=p;  
    p->kov=NULL;  
}
```

```
int menu(void)
```

```
{  
    int valasz;  
    system("cls");  
    cout<<"1 - Push()"<<endl;  
    cout<<"2 - Pop()"<<endl;  
    cout<<"3 - List()"<<endl;  
    cout<<"4 - Exit"<<endl;  
    cout<<"======"<<endl;  
    cout<<"valasztas: "; cin>>valasz;  
    return valasz;  
}
```

```
void pop(elem *&lista) //lista: listafej (elso)
```

```
{  
    elem *p, *q;  
    p=lista;  
    if (p->kov==NULL) cout<<endl<<"A lista ures!"<<endl;  
    else  
    if (p->kov==NULL) //csak egy elem van a listan  
    {  
        cout<<endl<<"Kiemelt elem: "<<p->adat<<endl;  
        delete p;  
        cout<<endl<<"A lista ures lett!"<<endl;  
    }  
    else  
    {  
        while (p->kov!=NULL) p=p->kov;//p elmegy az utolso elemig  
        //az utolso elem torlese:  
        q=lista;  
        while (q->kov!=p) q=q->kov;  
        q->kov=NULL; //ez az uj utolso  
        cout<<endl<<"Kiemelt elem: "<<p->adat<<endl;  
        delete p;  
    }  
}
```

```

void push(elem *lista)
{
    int szam;
    f>>szam;
    if (!f.eof()) hozzaad(lista, szam);
    else cout<<endl<<"Nincs tobb elem a file-ban!"<<endl;
}

void list(elem *lista)
{
    elem *p;
    p=lista->kov;
    if (lista->kov==NULL) cout<<endl<<"Nincs mit listazni."<<endl;
    else while (p!=NULL)
        {
            cout<<p->adat<<" ";
            p=p->kov;
        }
    cout<<endl;
    system("pause");
}

int main()
{
    cout << "LIFO lista letrehozasa" << endl;
    //Last in first out
    //be: 2 5 7 1 6 (push in)
    //ki: 6 1 stb (pop out)
    //A programot ugy kell megirni, hogy az uj adatok beirasa (hozzafuzese) csak az utolso
    //elem utan tortenhessen es egy elem kiemelese csak az utolso elemre vonatkozhasson.
    int MP, aktualis, ERTEK;
    elem *L; //ez lesz a LIFO lista (L: listafej)
    f>>ERTEK;
    init(L, ERTEK);
    do
    {
        MP=menu();
        switch (MP)
        {
            case 1: push(L); list(L); break;
            case 2: pop(L); list(L); break;
            case 3: list(L); break;
            case 4: cout<<"Program vege."; break;
            default: cout<<"Nincs ilyen menupont.";
                    system("pause"); break;
        }
    }while (MP!=4);
    f.close();
    return 0;
}

```