

Seminar 10

Ushtrimi 1

Nderton kodin e duhur ne C++ qe deklaron nje liste nje drejtimore e cila permban numra te plote dhe me ane te funksionit "Insert" i inserton elementet nga koka e listes.

```
#include <iostream>
#include <stdio.h>
#include <malloc.h>

using namespace std;

typedef struct element{
    int vlere;
    struct element *pas;
}liste;

liste* Insert(liste *l, int x)
{
    liste *temp;
    temp=new element;
    temp->vlere = x;
    temp->pas=l;
    l=temp;
    return l;
}
```

Ushtrimi 2

Nderton me ane te funksionit "afisho" kodin e duhur qe afishon elementet e listes ne rendin e kundert me ate te nderfutjes

```
void Afisho(liste *l){
    liste *elem;
    elem=l;
    while (elem!=NULL){
        cout<< elem->vlere <<" ";
        elem = elem->pas;
    }
}
```

Ushtrimi 3

Nderton funksionin GjatesiListe i cili llogarit gjatesine e nje liste

```
int GjatesiListe(liste *l)
{
    int gjatesi=0;
    while (l!=NULL){
        gjatesi++;
        l=l->pas;
    }
}
```

```

return gjatesi;
}

//funksioni qe llogarit gjatesine me thirrjen rekursive
int GjatesiListe_Rek(liste *l){

    if (l==NULL)
        return 0;
    else
        return 1+GjatesiListe_Rek(l->pas);
}

```

Ushtrimi 4

Ndertoni nje funksion qe kthen true nese nje element me vlere vl gjendet ne listed he false nese jo.

```

bool VIGjendet(liste *l, int vl){
    bool ugjet=false;
    int i=1;
    while ((!ugjet) && (l!=NULL))
    {
        ugjet=(l->vlere==vl);
        i++;
        if (!ugjet)
            l=l->pas;
    }
    return ugjet;
}

//funksioni rekursiv per gjetjen e nje elementi me vlere vl
bool VIGjendet_Rek(liste *l, int vl){
    if (l==NULL)
        return false;
    else if (vl==l->vlere)
        return true;
    else
        return VIGjendet_Rek(l->pas,vl);
}

```

Ushtrimi 5

Ndertoni nje funksion qe gjen elementin e k-te te listes

```

liste* ElementiK(liste *l, int k){
    int i=1;
    while ((i<k) && (l!=NULL)){
        i++;
        l=l->pas;
        if (i==k){
            cout<<"Elementi "<<k<<" gjendet ne liste\n";
        }
    }
}

```

```

    }
    return l;
}

```

Ushtrimi 6

Ndertoni nje funksion qe shton nje element te ri ne vendin e k-te te listes

```

liste* ShtimNePozK(liste* l, int x, int k)
{
    int i;
    liste* temp1;//elementi i ri qe do te shtojme
    liste* temp2;//var i perkohshem
    //temp1 = (element *)malloc(sizeof(element));
    temp1 = new element;
    temp1->vlere=x;
    i=1;
    //testojme nese k=1, koka e listes
    if (k==1)//shtimi ndodh ne koke te listes
    {
        temp1->pas=l;
        l=temp1;
    }
    else if ((k>1) && (l!=NULL)) //shtimi pas kokes se listes
    {
        temp2=l;//shenjues i perkohshem qe shenon ne koke te listes, ne liber var sh
        while (i<k-1)
        {
            i++;
            temp2=temp2->pas;
        }
        temp1->pas=temp2->pas;
        temp2->pas=temp1;
    }
    return l;
}

```

Ushtrimi 7

Ndertoni nje funksion qe shton nje element pas elementit te pare me vlere vl

```

liste* ShtimPasVl(liste *l, int vl, int x)
{
    liste *temp1;
    liste *temp2;
    temp2=l;
    while ((l->vlere!=vl) && (l!=NULL))
        l=l->pas;
    if (l->vlere==vl)
    {
        temp1=new element;
    }
}

```

```

        temp1->vlere=x;
        temp1->pas=l->pas;
        l->pas=temp1;
        l=temp2;
    }
    return l;
}

```

Ushtrimi 8

Ndertoni nje funksion qe heq nga lista elementin e k-te

```

liste* HeqjeElm(liste *l, int k)
{
    liste *temp1;//var qe shenon ne poz e k-te, ai qe duam te fshijme
    liste *temp2;//var qe bredh listen nga koka e saj dhe afishon gjith elementet pas fshirjes, ne
    liber var sh
    temp2=l;
    if (k==1)
    {
        temp1=l;
        l=l->pas;
        delete(temp1);
        cout<<"Elementi ne pozicionin "<<k<<" u hoq nga lista\n";
    }
    else
    {
        int i=1;
        while (i<k-1)
        {
            l=l->pas;
            i++;
        }
        temp1=l->pas;
        l->pas=temp1->pas;
        delete(temp1);
        l=temp2;
        cout<<"Elementi ne pozicionin "<<k<<" u hoq nga lista\n";
    }
    return l;
}

```

Ushtrimi 9

Ndertoni funksioni qe heq nga lista elementin e pare me vlere vl

```

liste * HeqElemPare(liste *l,int vl)
{
    liste *temp1;
    liste *temp2;
    temp2=l;
    if (l->vlere==vl)
    {

```

```

        temp1=l;
        l=l->pas;
        delete(temp1);
        cout<<"Elementi me vlere "<<vl<<" u hoq nga lista\n";
    }
    else
    {
        while (l->pas->vlere!=vl)
            l=l->pas;
        temp1=l->pas;
        l->pas=temp1->pas;
        delete(temp1);
        l=temp2;
        cout<<"Elementi me vlere "<<vl<<" u hoq nga lista\n";
    }
    return l;
}

```

Thirrja e te gjitha funksioneve te ndertuara me lart

```

int main()
{
    int vl;
    liste *L;
    //incializoj listen si liste boshe
    L=NULL;
    cout<<"Shkruani numrat qe doni te insertoni ne liste\n";
    for (int i=1;i<=7;i++){
        cin >> vl;
        L=Insert(L,vl);
    }
    cout<<"Po afishojme elementet e listes \n";
    Afisho(L);
    cout<<endl;
    cout<<"Gjatesiesia e Listes eshte: ";
    cout<<GjatesiListe_Rek(L)<<endl;
    cout<<endl;
    cout<<"Shkruani vleren qe doni te kerkoni ne liste\n";
    cin>>vl;
    if (VIGjendet(L,vl))
        cout<<"Vlera gjendet ne liste\n";
    else
        cout<<"Vlera nuk gjendet ne liste\n";
    cout<<VIGjendet_Rek(L,5);
    cout<<"\n\n";

    int k;
    cout<<"Shkruani nje vlere per k\n";
    cin>>k;
    ElementiK(L,k);

    cout<<"\n\n";
}

```

```
cout<<"Shkruani nr qe deshironi te shtoni ne poz e 4-te\n";
cin>>vl;
L=ShtimNePozK(L,vl,4);
cout<<endl;
Afisho(L);

cout<<endl;
cout<<"Shkruani nje vlere te plote\n";
cin>>vl;

L=ShtimPasVI(L,vl,4);
cout<<"U krye shtimi i elementit me vlere 4 pas elementit me vlere "<<vl<<"\n";
Afisho(L);

cout<<endl;
int poz_k;
cout<<"Shkruani nje vlere per k-ne\n";
cin>>poz_k;
L=HeqjeElm(L,poz_k);
Afisho(L);

cout<<endl;
cout<<"Heqje e elementit me vlere 3\n";
L=HeqElemPare(L,3);
Afisho(L);

return 0;
}
```