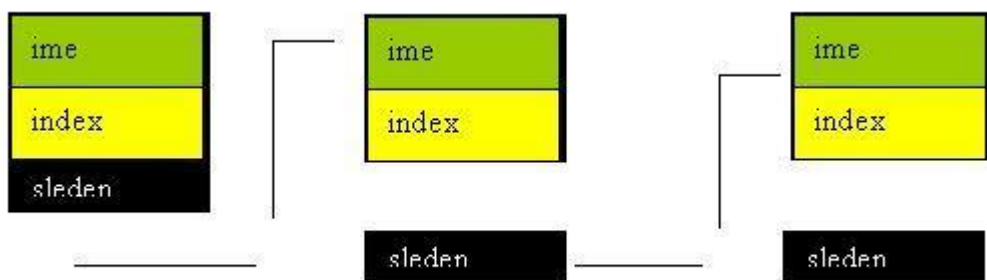


# Еднострано поврзани линеарни листи - креирање и додавање елементи

## Листи

Динамичката меморија се користи за сместување на големи и сложени типови на податоци, па покажувачките променливи служат за нивна манипулација. Тие можат да покажуваат кон локација од било кој тип на податок вклучувајќи ги и записите.

Запис е множество на податоци. Елементите на записот не се индексирани, туку секој податок има свој идентификатор (име) со кое се пристапува до него. На пример, записот за еден студент нека ги содржи податоците: име, број на индекс и следен. `ime` е текстуален податок, додека `index` е целоброен податок. `sleden` е покажувач од тип `рок`, кој е дефиниран како покажувач од тип `student`. Според дефиницијата на `student`, можеме да креираме листа од записи од ист тип, поврзани со употреба на покажувачи. Листата е илустрирана со следнава слика:



Поврзаната листа е една од основните динамички структури на податоци кои се употребуваат во програмирањето. Се состои од низа од јазли. Секој јазол има дел за сместување податоци и дел за сместување покажувач кој служи како врска кон наредниот јазол од низата. Поврзаните листи возможуваат вметнување и отстранување на јазли од било кое место во низата за константно време, но не дозволува пристап до јазлите по случаен избор. Односно, секвенцијалниот пристап е ефикасен, меѓутоа директниот не е бидејќи треба да се поминат сите елементи од листата за да се добијат потребните податоци.

Постојат неколку типови на поврзани листи: еднострано поврзани, двострано поврзани и кружни листи.

### Еднострано поврзани листи

Наједноставниот тип на поврзани листи се еднострано поврзаните листи кои имаат по еден покажувач за секој јазол, кој покажува кон наредниот јазол од листата, кон `null` вредност ако станува збор за последниот јазол од листата или кон празна листа. Анкер (`anchor`) на листата е покажувач кој покажува кон првиот јазол. За да се промени редоследот на листата доволно е само да се промени вредноста на покажувачот.



Секој јазол од листата има свој *следбеник* (освен последниот) и свој *предходник* (освен првиот). *Должината* на поврзана листа се изразува преку бројот на јазли од кои е составена. Листа која не содржи јазли се нарекува *празна листа*.

Пр. Да се креира еднострано поврзана листа со 10 елементи цели броеви и да се испечатат само парните броеви

```
#include <iostream>
using namespace std;

struct jazol
{
    int x;
    jazol *pok;
};

int main()
{
    jazol *poc,*nov, *sleden;
    nov=new jazol;
    cout<<"vnesi broj "<<endl;
    cin>>nov->x;
    nov->pok=NULL;
    poc=nov;
    for(int i=1;i<=9;i++)
    {
        cout<<"vnesi broj "<<endl;
        sleden=new jazol;
        cin>>sleden->x;
        sleden->pok=NULL;
        nov->pok=sleden;
        nov=sleden;
    }
    while(poc!=NULL)
    {
        if (poc->x%2==0)
            cout<<poc->x<<" ";
        poc=poc->pok;
    }
    return 0;
}
```

*//се креира почетен елемент на листата*

*//внеси вредност на елементот*

*//првиот елемент е единствен во листата и покажувачот*

*//првиот елемент е почеток на листата*

*//се креира нов елемент на листата*

*//внеси вредност на елементот*

*//новиот елемент е последен во листата па неговиот*

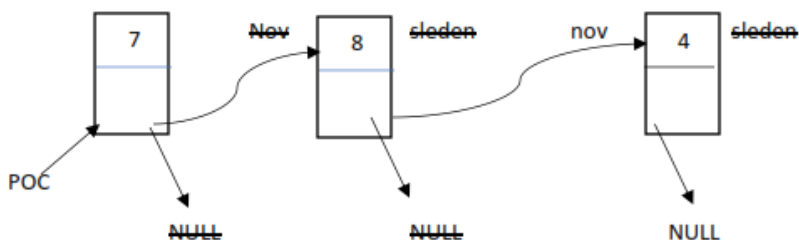
*//покажувачот од претходниот елемент од листата*

*//претходниот елемент има вредност на последниот за да*

*може постапката да се повторува се додека треба да се додадат нови елементи. Додавањето на нов елемент*

*е на крајот на листата*

*//печатење на парните елементи од листата*



Пр. Да се креира еднострано поврзана листа со n елементи цели броеви. Додавање на елемент е на почеток на листата. Да се испечати листата

```
#include <iostream>
using namespace std;

struct jazol
{
    int x;
    jazol *pok;
};

int main()
{
    jazol *poc, *nov, *sleden;
    int n;
    nov=new jazol; //се креира почетен елемент на листата
    nov->pok=NULL; //првиот елемент е единствен во листата и покажувачот
// има вредност NULL
    poc=nov; //првиот елемент е почеток на листата
    cout<<"vnesi broj na elementi na listata "<<endl;
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        cout<<"vnesi broj "<<endl;
        sleden=new jazol; //се креира нов елемент на листата
        cin>>sleden->x; //внеси вредност на елементот
        sleden->pok=poc; //новиот елемент е прв па неговиот покажувачсе насочува
// да покажува на почеток
        poc=sleden; //pocetok има вредност на последниот елемент
    }
    while(poc->pok!=NULL)
    {
        //печатење на елементите од листата
        cout<<poc->x<<" ";
        poc=poc->pok;
    }
    return 0;
}
```

Пр. Да се креира еднострано поврзана листа со n елементи цели броеви и да се испечатат броевите

```
#include <iostream>
using namespace std;
```

```

struct jazol
{
    int x;
    jazol *pok;
};

int main()
{
    jazol *poc,*nov, *sleden;
    nov=new jazol;
    int n;
    cout<<"vnesi broj "<<endl;
    cin>>nov->x;
    nov->pok=NULL;
    poc=nov;

    cout<<"vnesi broj na elementi "<<endl;
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        cout<<"vnesi broj "<<endl;
        sleden=new jazol;
        cin>>sleden->x;
        sleden->pok=NULL;
        nov->pok=sleden;
        nov=sleden;

        while(poc!=NULL)
        {
            cout<<poc->x<<" ";
            poc=poc->pok;
        }
        return 0;
    }
}

```

*//се креира почетен елемент на листата*

*//внеси вредност на елементот*

*//првиот елемент е единствен во листата и покажувачот има вредност NULL*

*//првиот елемент е почеток на листата*

*//се креира нов елемент на листата*

*//внеси вредност на елементот*

*//новиот елемент е последен во листата па неговиот покажувач покажува на NULL*

*//покажувачот од претходниот елемент од листата покажува на новиот елемент*

*//претходниот елемент има вредност на последниот за да може постапката да се повторува се додека треба да се додадат нови елементи. Додавањето на нов елемент е на крајот на листата*

Пр. Да се креира еднострано поврзана листа со n елементи цели броеви. Елемент се додава на било кое место во листата. Да се внеси елементот што се додава и да се испечати новата листа

```

#include <iostream>
using namespace std;

```

```

struct jazol
{
    int x;

```

```

    jazol *pok;
};

int main()
{
    jazol *poc,*p,*nov, *sleden;
    int n,l,k,t;
    nov=new jazol; //се креира почетен елемент на листата
    nov->pok=NULL; //првиот елемент е единствен во листата и покажувачот има вредност NULL
    poc=nov; //првиот елемент е почеток на листата
    cout<<"vnesi broj na elementi na listata "<<endl;
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        cout<<"vnesi broj "<<endl;
        sleden=new jazol; //се креира нов елемент на листата
        cin>>sleden->x; //внеси вредност на елементот
        sleden->pok=poc; //новиот елемент е прв па неговиот покажувач се насочува да покажува на почеток
        poc=sleden; //pocetok има вредност на последниот елемент
    }

    while(poc->pok!=NULL)
    {
        //печатење на елементите од листата
        cout<<poc->x<<" ";
        poc=poc->pok;
    }
    poc=sleden;
    p=sleden;
    cout<<"vnesi go el sto go dodavas"<<endl;
    cin>>l;
    cout<<"vnesi po koj el go dodavas"<<endl;
    cin>>k;
    while(p->pok!=NULL)
    {
        if (k==poc->x) //go bara elementot
        {
            sleden=new jazol; //koga ke go najdi kreira nov jazol
            sleden->x=l; //negova soдрzina e elementot sto go dodavas

            sleden->pok=poc->pok; //pokazuvacot od sledniot element pokazuva na elementot na koj
            poc=>pok=sleden; //pokazuvacot od poc pokazuva na jazolot koj e kreiran i se povрziva
            break; //prekin za da ne bara ponatamu
        }
        else

```

```

    poc=poc->pok; //ako ne e najden elementot, preminuva na sleden element
}

cout<<"novata niza e "<<endl;
while(p->pok!=NULL)
{
    cout<<p->x<<" ";
    p=p->pok;
}
return 0;
}

```

### **Задачи за домашна работа:**

1. Да се напише програма со која се креира еднострано поврзана листа со  $n$  елементи и да се додаде елемент по првиот парен елемент во листата.
2. Да се напише програма со која се креира еднострано поврзана листа со  $n$  елементи и да се печатат елементите кои се делат со 3
3. Да се напише програма со која се креира еднострано поврзана листа со  $n$  елементи и да се додаде елемент по вториот елемент од листата
4. Да се напише програма со која се креира еднострано поврзана листа со  $n$  елементи и да се печатат елементите кои припаѓаат на втората десетка.

**Програмите што се претходно изработени да се напишат со користење на функции за креирање, додавање и печатење на листа.**

*Прашања поврзани со наставните единици може да се испраќаат на email: [anetastojceska@gmail.com](mailto:anetastojceska@gmail.com)*