Universiteti i Prishtinës

Fakulteti i Inxhinierisë Elektrike dhe Kompjuterike

Algoritmet dhe struktura e të dhënave

Vehbi Neziri

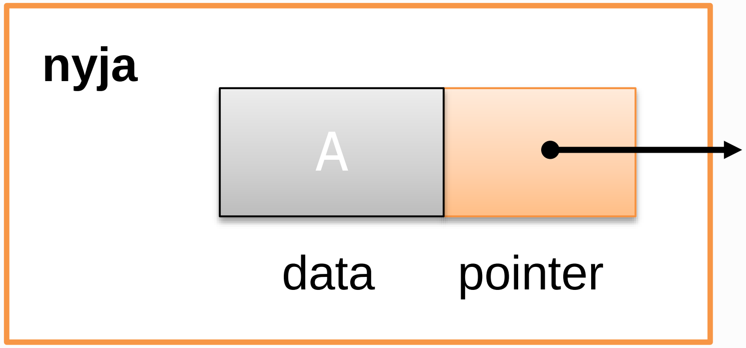
FIEK, Prishtinë 2015

- Listat e lidhura
  - Njëfishe
  - Dyfishe
  - Cirkulare

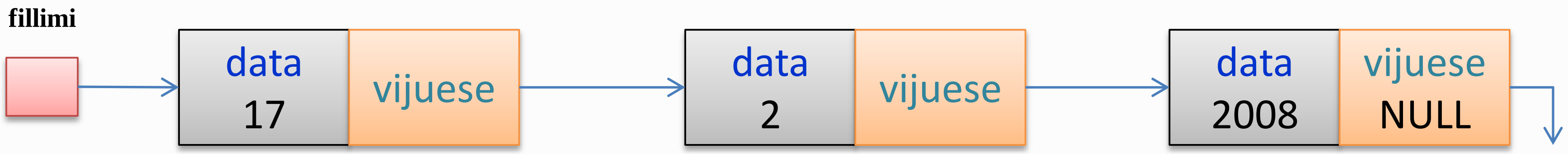
- Veprimet me lista
  - Mbushja e një nyje
  - Mbushja e dy nyjave
  - Mbushja e shumë nyjave
  - Shfaqja e adresës dhe vlerës së nyjave

Listat e lidhura

- Seri e nyjave të lidhura
  - Secila nyje (përjashtuar të fundit) përmban adresën e nyjës tjetër.
- Secila nyje përmban
  - Pjesën e të dhënave (çfarë do tipi)
  - Pointerin në nyjen tjetër në listë

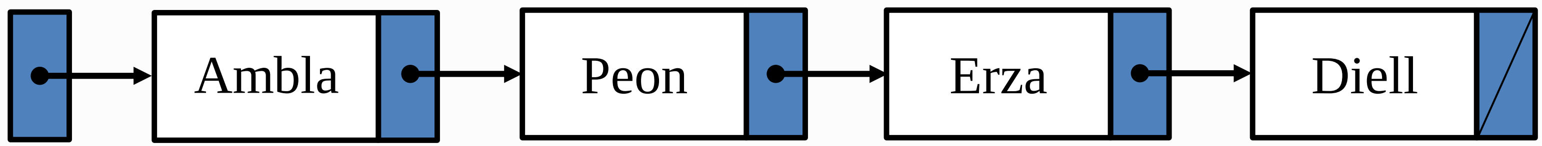


- Kreu (e para)
  - Pointeri tek nyja e parë.
- Nyja e fundit tregon NULL
- Lista e lidhura mund të deklarohen si strukturë ose si klasë.

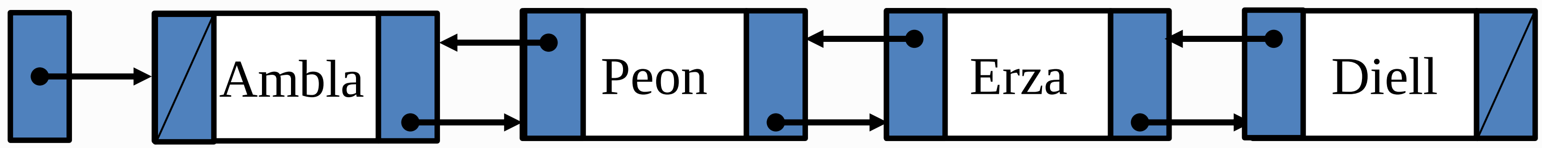


Listat e lidhura

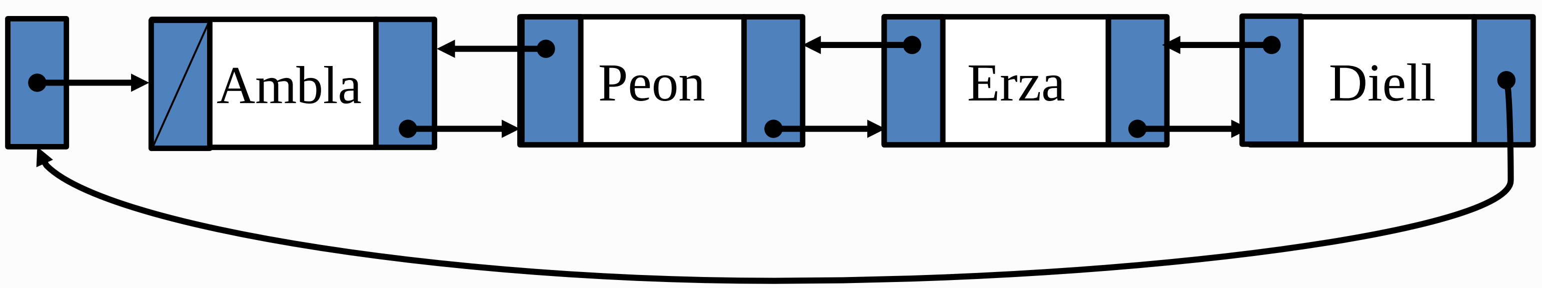
- Lista njëfishe



- Lista dyfishe



- Lista cirkulare



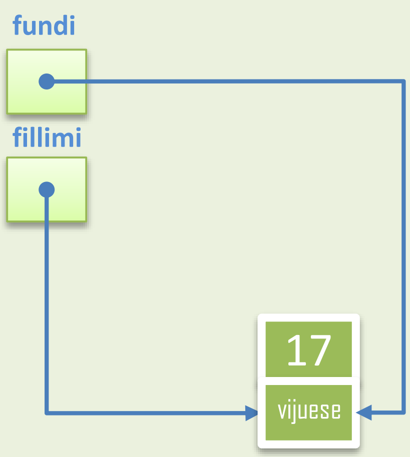
Ndërfutja e nyjave të reja

- Mundësitë e ndërfutjes së nyjës së re
  a) Ndërfut në listën e zbrazur
  b) Ndërfut në krye (front)
  c) Ndërfut në fund (back)
  d) Ndërfut në mes (middle)

- Konsiderohet se janë vetëm dy situate
  - Ndërfutja si nyje e parë (rasti a dhe b)
  - Ndërfutja në mes ose në fund (rasti c dhe d)

Shembull 8.1

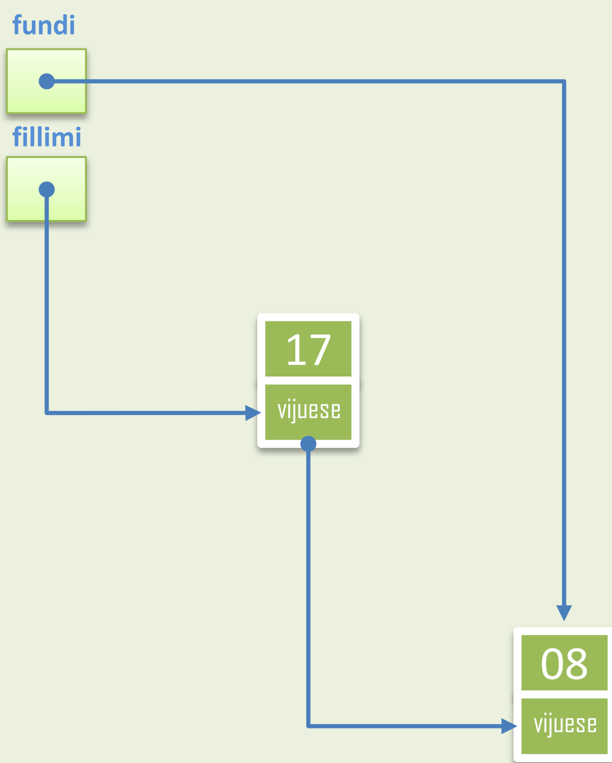
- Të shkruhet programi në të cilin definohet një strukturë për nyjën. Në funksionin main të deklarohet një pointer i nyjës përmes të cilit mbushet nyja me vlerë dhe më pas të shtypet vlera e kësaj nyje.




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

struct Nyja
{
    double e_dhena; // pjesa e të dhënave
    Nyja* vijuese;  // pointer në nyjën tjetër
};

int main()
{
    Nyja *n;    //Pointer në strukturën nyja
    n = new Nyja;
    n->e_dhena = 17;
    n->vijuese = NULL;

    cout << "Vlera e nyjes : "
        << n->e_dhena << endl;
    cout << "Nyja vijuese: "
        << n->vijuese << endl;
    cout << endl;

    system("Pause");
    return 0;
}
```

Shembull 8.2

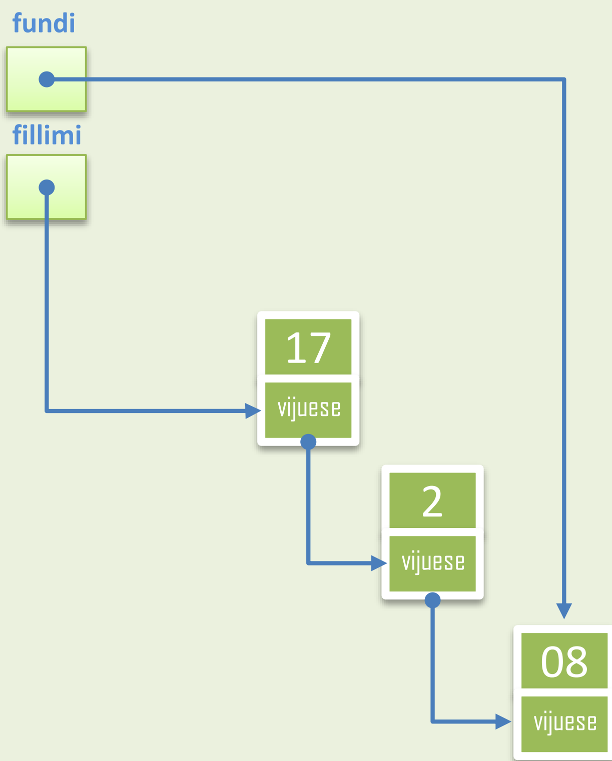
- Të zgjerohet shembulli paraprak dhe të mundësohet shtimi edhe i një nyje tjetër. Në fund të shtypet adresa dhe vlera e të dy nyjave.




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

struct Nyja
{
    double e_dhena; // pjesa e të dhënave
    Nyja* vijuese;  // pointer në nyjën tjetër
};

int main()
{
    Nyja *n, *e_para=NULL, *e_fundit=NULL;
    n = new Nyja;
    n->e_dhena = 17;
    n->vijuese = NULL;
    e_para = n;
    e_fundit = n;

    n = new Nyja;
    e_fundit->vijuese = n;
    n->e_dhena = 2;
    n->vijuese = NULL;
    e_fundit = n;

    cout << "\nMbushja e dy nyjave perfundoi"
         << "\n\nFillimi: " << e_para
         << " Fundi: " << e_fundit
         << "\n\nPermbajtja e listes";

    n = e_para;
    cout << "\n\nAdresa   Vlera Adresa vijuese\n\n"
        << "-------------------------------------------\n"
        << setw(8) << n
        << setw(5) << n->e_dhena
        << setw(9) << n->vijuese << endl;
    n = n->vijuese; // adresa që gjendet te vijuese ruhet në n
    cout << setw(8) << n
        << setw(5) << n->e_dhena
        << setw(9) << n->vijuese
        << endl;

    system("Pause");
    return 0;
}
```

Shembull 8.3

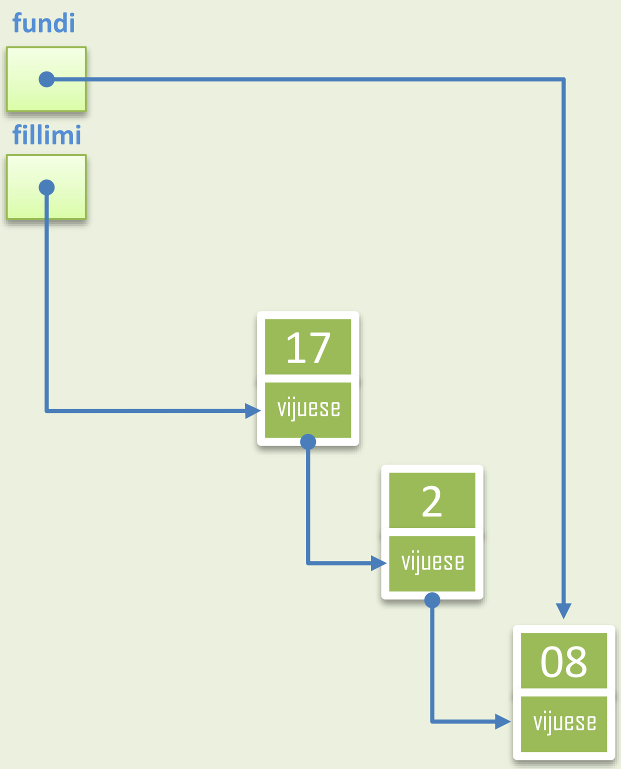
- Të zgjerohet shembulli paraprak dhe dhe të mundësohet shtimi i **n** nyjave. Numri i nyjave dhe vlerat për secilën nyje të lexohen përmes tastiere.




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

struct Nyja
{
    double e_dhena; // pjesa e të dhënave
    Nyja* vijuese;  // pointer në nyjën tjetër
};

int main()
{
    Nyja *n, *e_para = NULL, *e_fundit = NULL;
    int vlera, nr_nyjave;
    cout << "Numri i nyjave: ";
    cin >> nr_nyjave;

    for (int i = 1; i <= nr_nyjave; i++)
    {
        n = new Nyja; //nyje e re
        cout << "Vlera e nyjes " << i << ": ";
        cin >> vlera;
        n->e_dhena = vlera;
        n->vijuese = NULL;
        if (e_para == NULL)
            e_para = n;
        else
            e_fundit->vijuese = n;
        e_fundit = n;
    }

    system("Pause");
    return 0;
}
```

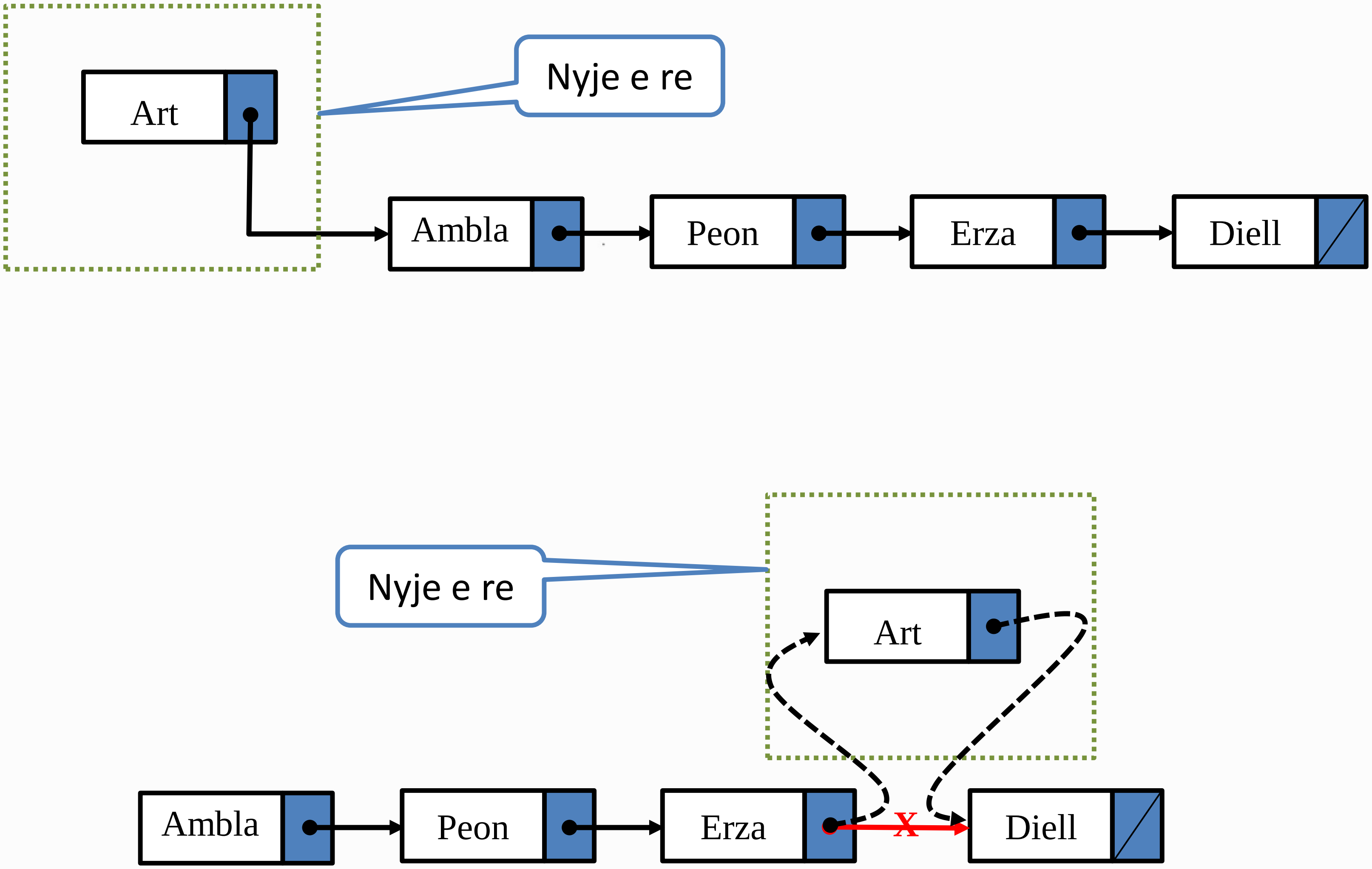
Shembull 8.4

- Të zgjerohet shembulli paraprak dhe përmes funksionit **shfaqListen** të shtypet përmbajtja e listës.




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

struct Nyja
{
    double e_dhena; // pjesa e të dhënave
    Nyja* vijuese;  // pointer në nyjën tjetër
};

void shfaqListen(Nyja *nyja1)
{
    Nyja *n = nyja1;

    cout << "\n\n\nPermbajtja e listes:"
        << "\n\n  Adresa     Vlera   Adresa vijuese\n"
        << "-----------------------------------------\n";

    while (n != 0)
    {
        cout << setw(10) << n
            << setw(5) << n->e_dhena
            << setw(15) << n->vijuese
            << endl;
        n = n->vijuese;
    }
    cout << endl;
}

int main()
{
    Nyja *n, *e_para = NULL, *e_fundit = NULL;
    int vlera, nr_nyjave;
    cout << "Numri i nyjave: ";
    cin >> nr_nyjave;

    for (int i = 1; i <= nr_nyjave; i++)
    {
        n = new Nyja;  //nyje e re
        cout << "Vlera e nyjes " << i << ": ";
        cin >> vlera;
        n->e_dhena = vlera;
        n->vijuese = NULL;
        if (e_para == NULL)
            e_para = n;
        else
            e_fundit->vijuese = n;
        e_fundit = n;
    }

    shfaqListen(e_para);

    system("Pause");
    return 0;
}
```

Ndërfutja në fillim dhe në mes



Shembull 8.5

- Të modifikohet shembulli paraprak dhe duke përdorur funksionin **shto_ne_fillim()** të shtohen nyjat e reja në fillim. Shtypja e përmbajtjes së listës të bëhet përmes funksionit **shfaqListen**().

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

struct Nyja
{
    double e_dhena; // pjesa e të dhënave
    Nyja* vijuese;  // pointer në nyjën tjetër
} *e_para;

void shfaqListen(Nyja *nyja1)
{
    Nyja *n = nyja1;
    cout << "\n\n\nPermbajtja e listes:"
        << "\n\n  Adresa     Vlera   Adresa vijuese\n"
        << "----------------------------------------\n";
    while (n != 0)
    {
        cout << setw(10) << n
            << setw(5) << n->e_dhena
            << setw(15) << n->vijuese
            << endl;
        n = n->vijuese;
    }
    cout << endl;
}

void shto_ne_fillim()
{
    int vlera;
    cout << "\n\nVlera e nyjes: ";
    cin >> vlera;
    Nyja *n;
    n = new Nyja;
    n->e_dhena = vlera;
    n->vijuese = e_para;
    e_para = n;
}

int main()
{
    char pyet = 'P'; int i = 1;

    for (int i = 1; i < 4; i++)
    {
        shto_ne_fillim();
    }
    shfaqListen(e_para);

    system("pause");
    return 0;
}
```

Shembull 8.6

- Të modifikohet shembulli paraprak dhe të shtohet funksioni **shto_pas_nyjes()** i cili nyjat e reja i shton pas nyjës se zgjedhur. Shtypja e përmbajtjes së listës të bëhet përmes funksionit **shfaqListen**() para dhe pas ndërfutjes së nyjave të caktuara.

```cpp

void shto_pas_nyjes()
{
    int vlera;
    int nyja;
    cout << "\n\nPas ciles nyje deshiron ta shtosh nyjen e re?: ";
    cin >> nyja;
    cout << "\n\nVlera e nyjes: ";
    cin >> vlera;

    Nyja *n, *e_re;
    n = e_para;

    for (int i = 1; i<nyja; i++)
    {
        n = n->vijuese;
        if (n == NULL)
        {
            cout << "\nJane me pak se " << nyja << " nyje.";
            return;
        }
    }

    e_re = new Nyja;
    e_re->e_dhena = vlera;
    e_re->vijuese = n->vijuese;
    n->vijuese = e_re;
}

int main()
{
    char pyet = 'P'; int i = 1;

    for (int i = 1; i <= 4; i++)
    {
        shto_ne_fillim();
    }

    shfaqListen(e_para); //shfaq para nderfutjes se nyjes se re

    shto_pas_nyjes();

    shfaqListen(e_para);    //shfaq pas nderfutjes se nyjes se re

    system("pause");
    return 0;
}
```

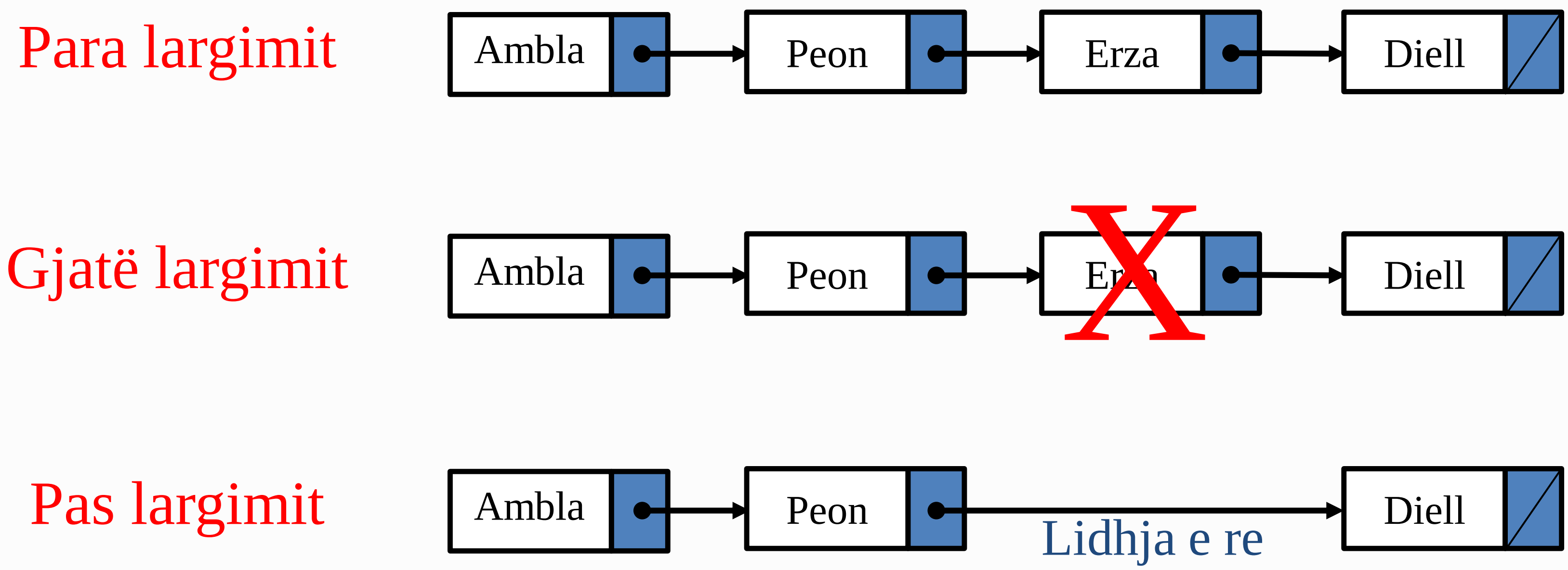
Shembull 8.7

- Të modifikohet shembulli 8.5 dhe të shtohet një funksion i cili numëron sa nyje i ka lista.

```
int numero_nyjat()
{
    Nyja *n;
    int c = 0;
    for (n = e_para; n != NULL; n = n->vijuese)
        c++;
    return  c;
}

int main()
{
    char pyet = 'P'; int i = 1;

    for (int i = 1; i < 4; i++)
    {
        shto_ne_fillim();
    }
    shfaqListen(e_para);
    cout << "\nNumri i nyjave eshte: " << numero_nyjat();

    system("pause");
    return 0;
}
```

Largimi i nyjave

- Largimi ose heqja e nyjave bëhet në disa mënyra
  - Largimi nga fillimi
  - Largimi nga fundi
  - Largimi I nyjës specifike
    - Sipas vlerës
    - Sipas pozitës



Shembull 8.8

- Të modifikohet shembulli paraprak dhe të shtohet edhe funksioni për largimin e nyjave sipas vlerës së nyjës.

```
void largo_nyjen_sipas_vleres(Nyja *nyja1)
{
    int vlera;
    cout << "\n\nSheno vleren qe deshiron te largosh: ";
    cin >> vlera;

    Nyja *n, *r;
    n = nyja1;

    if (n->e_dhena == vlera)
    {
        nyja1 = n->vijuese;
        delete n;
        return;
    }

    r = n;
    while (n != NULL)
    {
        if (n->e_dhena == vlera)
        {
            r->vijuese = n->vijuese;
            delete n;
            return;
        }

        r = n;
        n = n->vijuese;
    }
    cout << "\nNyja me vlere " << vlera << " nuk u gjend.";
}
```