

T04 t siny s. Susietas s ra as (linked list)

1 ak. val.

Temos klausimai

1. Susieto sąrašo mazgo klasė (`LinkedListNode (T)` (`System.Collections.Generic`)).
2. Susieto sąrašo klasė (`LinkedList (T)` `class` (`System.Collections.Generic`)).



LinkedListNode<T> Class (System.Collections.Generic)

Mazgas `LinkedListNode<T>`

1/4

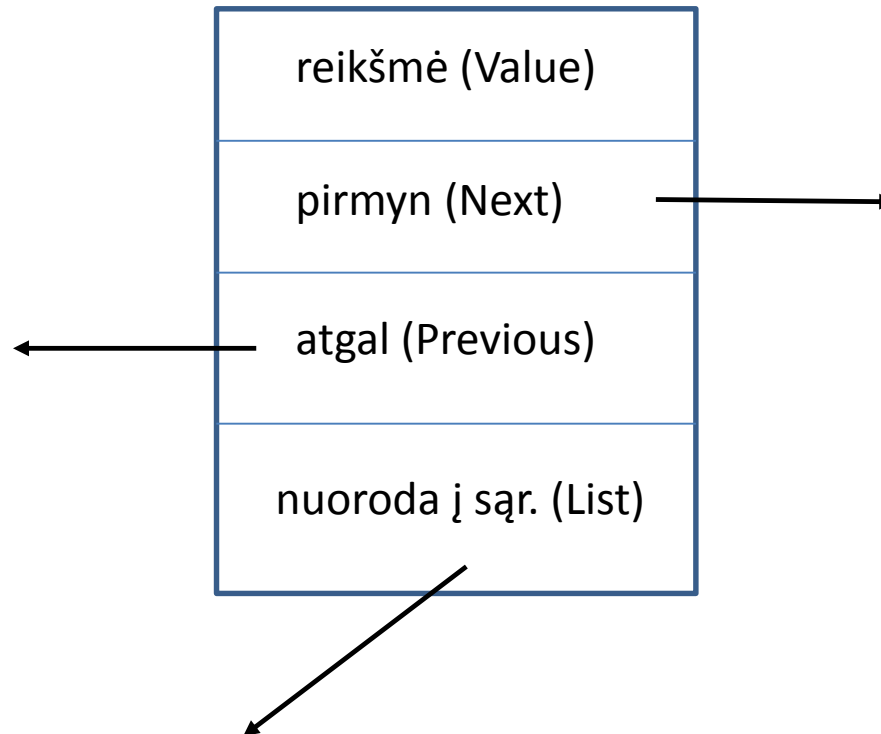
Mazgas apibūdina susieto sąrašo `LinkedList<T>` elementą. Ši klasė nepaveldima.

Kiekvieną mazgą aprašo:

- duomenys (reikšmė) - value;
- nuoroda į susietą sąrašą, kuriam jis priklauso - list;
- nuoroda į sekantį sąrašo elementą - next;
- nuoroda į buvusį sąrašo elementą - previous.

Mazgas `LinkedListNode<T>`

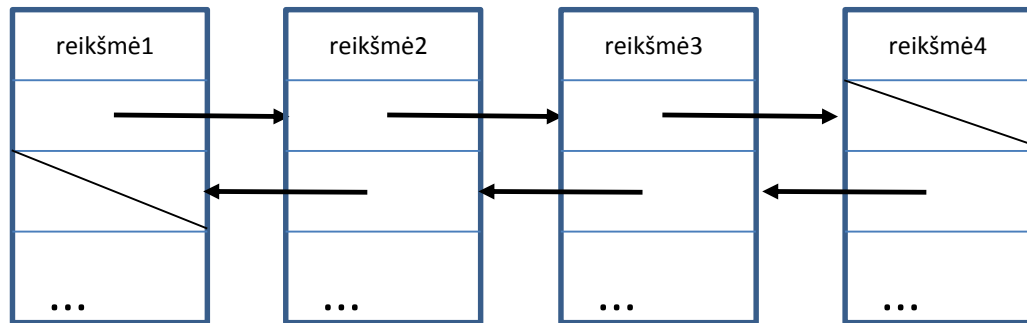
2/4



Mazgas `LinkedListNode<T>`

3/4

Susieto sąrašo mazgai `LinkedListNode<T>`
sudaro susietą sąrašą `LinkedList<T>`:



Mazgas `LinkedListNode<T>`

4/4

Susieto sąrašo mazgas `LinkedListNode<T>` turi:

- 1 konstruktorių.
- 4 savybes (properties).
- 4 metodus.

Mazgas `LinkedListNode<T>` aprašas

Aprašas:

```
LinkedListNode<T> Pavadinimas =  
    new LinkedListNode<T> (reikšmė);
```

Sukuriamas naujas mazgas **Pavadinimas**, kuriame yra įrašyta nurodyta **reikšmė**.

```
LinkedListNode<String> mazg =  
    new LinkedListNode<String>("Paskaita");  
Console.WriteLine("Sukurto mazgo mazg reikšmė:{0}", mazg.Value);  
Console.WriteLine();
```

```
Sukurto mazgo mazg reikšmė: Paskaita  
Press any key to continue . . .
```


Dažniausiai naudojami susieto sąrašo mazgo `LinkedListNode<T>` metodai

| Metodas arba savybė | Aprašas |
|-------------------------------|---|
| <code>Equals(objektas)</code> | Grąžina true, jei nurodytas objektas lygus nagrinėjamam objektui, priešingu atveju – false. (Paveldėta iš <code>Object</code>) |
| <code>List</code> | Grąžina nuorodą į susietą sąrašą, kuriam priklauso mazgas, arba <code>null</code> . |
| <code>Next</code> | Grąžina nuorodą į sekantį sąrašo narį arba <code>null</code> . |
| <code>Previous</code> | Grąžina nuorodą į prieš tai esantį sąrašo narį arba <code>null</code> . |
| <code>Value</code> | Grąžina mazge įrašytą reikšmę. |

Susieto sąrašo mazgo

LinkedListNode<T>

pavyzdžiai (1/6)

```
public class Asmuo
{
    public string vardas { get; set; }
    public int amžius { get; set; }
    public Asmuo(string vardas, int amžius) // Konstruktorius
    {
        this.vardas = vardas;
        this.amžius = amžius;
    }
    public override string ToString()
    {
        return this.vardas + " " + this.amžius;
    }
    public override int GetHashCode()
    {
        return base.GetHashCode();
    }
}
```

Susieto sąrašo mazgo `LinkedListNode<T>` pavyzdžiai (2/6)

```
// spausdina susieto sąrašo reikšmes (Asmuo).  
public static void Spausdinti(LinkedList<Asmuo> sąrašas)  
{  
    foreach (Asmuo elem in sąrašas)  
    {  
        Console.WriteLine("    {0}", elem);  
    }  
    Console.WriteLine();  
}
```

Susieto sąrašo mazgo `LinkedListNode<T>` pavyzdžiai (3/6)

```
// nustato mazgo savybes
public static void Savybes(LinkedListNode<Asmuo> mazgas)
{
    Console.WriteLine("    Nagrinėjamo mazgo reikšmė: {0}", mazgas.Value);
    if (mazgas.List == null) Console.WriteLine("    Mazgas neprijungtas.");
    else Console.WriteLine("    Mazgas yra susietame sąraše.");

    if (mazgas.Previous == null) Console.WriteLine("    Nėra elemento prieš jį.");
    else Console.WriteLine("    Prieš šį mazgą esančio mazgo reikšmė: {0}",
        mazgas.Previous.Value);

    if (mazgas.Next == null) Console.WriteLine("    Nėra elemento už jo.");
    else Console.WriteLine("    Už šio mazgo esančio mazgo reikšmė: {0}",
        mazgas.Next.Value);

    Console.WriteLine();
}
```

Susieto sąrašo mazgo `LinkedListNode<T>` pavyzdžiai (4/6)

```
// Sukuria ir inicializuoja susietą sąrašą
LinkedList<Asmuo> antras = new LinkedList<Asmuo>();
Asmuo stud = new Asmuo("Jonas", 25);
Asmuo stud1 = new Asmuo("Petras", 26);
Asmuo stud2 = new Asmuo("Juozas", 38);
Asmuo stud3 = new Asmuo("Kazys", 25);
antras.AddFirst(stud);
antras.AddLast(stud1);
antras.AddLast(stud2);
antras.AddLast(stud2);
Console.WriteLine("    Susietas sąrašas antras");
Console.WriteLine("Sąrašo narių kiekis:    {0}", antras.Count);
Console.WriteLine("    Reikšmės:");
Spausdinti(antras);
```

```
Susietas sąrašas antras
Sąrašo narių kiekis:    4
Reikšmės:
Jonas    25
Petras   26
Juozas   38
Juozas   38
Press any key to continue . . .
```

Susieto sąrašo mazgo `LinkedListNode<T>` pavyzdžiai (5/6)

```
// Sukuria ir inicializuoja mazgą
LinkedListNode<Asmuo> mazgas = new LinkedListNode<Asmuo>(stud);
Console.WriteLine("Sukurtas mazgas");
Savybes(mazgas);
// Sukuria ir inicializuoja susietą sąrašą
LinkedList<Asmuo> pirmas = new LinkedList<Asmuo>();
pirmas.AddLast(mazgas);
Console.WriteLine("Susietas vieno mazgo sąrašas");
Savybes(mazgas);
// Susietas sąrašas po papildymo
pirmas.AddFirst(stud1);
pirmas.AddLast(stud2);
Console.WriteLine("Papildytas susietas sąrašas");
Savybes(mazgas);
```

Susieto sąrašo mazgo `LinkedListNode<T>` pavyzdžiai (6/6)

stud1

stud2

Petras 26

Juozas 38

```
Sukurtas mazgas
  Nagrinėjamo mazgo reikšmė: Jonas 25
  Mazgas neprijungtas.
  Nėra elemento prieš jį.
  Nėra elemento už jo.

Susietas vieno mazgo sąrašas
  Nagrinėjamo mazgo reikšmė: Jonas 25
  Mazgas yra susietame sąraše.
  Nėra elemento prieš jį.
  Nėra elemento už jo.

Papildytas susietas sąrašas
  Nagrinėjamo mazgo reikšmė: Jonas 25
  Mazgas yra susietame sąraše.
  Prieš šį mazgą esančio mazgo reikšmė: Petras 26
  Už šio mazgo esančio mazgo reikšmė: Juozas 38

Press any key to continue . . .
```



LinkedList<T> Class *(System.Collections.Generic)*

Susietas sąrašas

`LinkedList<T>` 1/2

Susietas sąrašas `LinkedList<T>` - dvipusis sąrašas.

Jis naudoja `ICollection` sąsają.

Visi sąrašo mazgai yra `LinkedListNode<T>` klasės objektai.

Mazgą galima išmesti iš sąrašo, jį iš naujo įterpti į tą patį ar kitą sąrašą.

Sąrašo mazgų reikšmės gali kartotis.

Jei sąrašas tuščias, *First* ir *Last* savybės yra *null*.

Susietas sąrašas

LinkedList<T> 2/2

Susietas sąrašas **LinkedList<T>** turi:

- 3 užklotus konstruktorius.
- 3 savybes (properties).
- 26 metodų.
- 7 sąsajos realizacijos (Explicit Interface Implementations).
- Didelę aibę užklotų metodų (>130).

Susieto sąrašo `LinkedList<T>` aprašas

Susieto sąrašo `LinkedList<T>` aprašai:

1. `LinkedList<T>` Pavadinimas =
`new LinkedList<T> ()`;

Sukuriamas *tuščias* (Count = 0) susietas sąrašas.

2. `LinkedList<T>` Pavadinimas1 =
`LinkedList<T> (Pavadinimas)`;

Sukuriamas naujas susietas sąrašas **Pavadinimas1**, kuriame yra bet kokio kito tipo sąrašo **Pavadinimas**, naudojantčio **IEnumerable** sąsają, elementai.

Dažniausiai naudojami susieto sąrašo **LinkedList<T>** metodai 1/2

| Metodas arba savybė | Aprašas |
|-----------------------------|---|
| AddAfter(mazgas, reikšmė) | Įterpia į sąrašą naują mazgą, turintį reikšmę reikšmė , už nurodyto mazgo. |
| AddAfter(mazgas1, mazgas2) | Įterpia į sąrašą naują mazgą mazgas2 už nurodyto mazgo mazgas1 . |
| AddBefore(mazgas, reikšmė) | Įterpia į sąrašą naują mazgą, turintį reikšmę reikšmė , prieš nurodytą mazgą. |
| AddBefore(mazgas1, mazgas2) | Įterpia į sąrašą naują mazgą mazgas2 prieš nurodytą mazgą mazgas1 . |
| AddFirst(reikšmė) | Įterpia į sąrašo pradžią naują mazgą, turintį reikšmę reikšmė . |
| AddFirst(mazgas) | Įterpia į sąrašo pradžią mazgą mazgas . |
| AddLast(reikšmė) | Įterpia į sąrašo pabaigą naują mazgą, turintį reikšmę reikšmė . |
| AddLast(mazgas) | Įterpia į sąrašo pabaigą mazgą mazgas . |
| Clear() | Pašalina visus sąrašo elementus. |
| Contains(reikšmė) | Grąžina true, jei ieškoma reikšmė yra, priešingu atveju – false. |
| CopyTo(Masyvas, indeksas) | Kopijuoja sąrašo reikšmes į atitinkamo tipo vienmatį masyvą, pradedant nurodytu masyvo indeksu. |
| Count | Savybė, kuri grąžina sąrašo elementų skaičių. |

Dažniausiai naudojami susieto sąrašo **LinkedList<T>** metodai 2/2

| Metodas arba savybė | Aprašas |
|---------------------|---|
| Equals(objektas) | Grąžina true, jei nurodytas objektas lygus nagrinėjamam objektui, priešingu atveju – false. (Paveldėta iš Object) |
| Find(reikšmė) | Randa pirmą sąrašo mazgą, turintį nurodytą reikšmę. |
| FindLast(reikšmė) | Randa paskutinį sąrašo mazgą, turintį nurodytą reikšmę. |
| First | Savybė, gąžinanti pirmą sąrašo mazgą. |
| GetEnumerator() | Grąžina sąrašo enumeratorių. |
| Count | Savybė, kuri grąžina žodyno elementų skaičių. |
| Last | Savybė, gąžinanti paskutinį sąrašo mazgą. |
| Remove(reikšmė) | Pašalina iš sąrašo pirmą mazgą, turintį nurodytą reikšmę. |
| Remove(mazgas) | Pašalina iš sąrašo nurodytą mazgą. |
| RemoveFirst() | Išmeta iš sąrašo mazgą, esantį sąrašo pradžioje. |
| RemoveLast() | Išmeta iš sąrašo mazgą, esantį sąrašo pabaigoje. |

Susieto sąrašo `LinkedList<T>` pavyzdžiai (1/18)

```
public class Asmuo
{
    public string vardas { get; set; }
    public int amžius { get; set; }
    public Asmuo(string vardas, int amžius) // Konstruktorius
    {
        this.vardas = vardas;
        this.amžius = amžius;
    }
    public override string ToString()
    {
        return this.vardas + " " + this.amžius;
    }
    public override int GetHashCode()
    {
        return base.GetHashCode();
    }
}
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (2/18)

```
// spausdina susieto sąrašo reikšmes (string).
public static void Spausdinti(LinkedList<string> sąrašas)
{
    foreach (string elem in sąrašas)
        Console.WriteLine("    {0,-10}", elem);
    Console.WriteLine();
}

// spausdina susieto sąrašo reikšmes (Asmuo).
public static void Spausdinti(LinkedList<Asmuo> sąrašas)
{
    foreach (Asmuo elem in sąrašas)
        Console.WriteLine("    {0}", elem);
    Console.WriteLine();
}
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (3/18)

```
// Sukuria ir inicializuoja susietą sąrašą
LinkedList<string> pirmas = new LinkedList<string>();
pirmas.AddFirst("vienas");      // įterpia naują mazgą pradžioje
pirmas.AddLast("penki");       // įterpia naują mazgą gale
pirmas.AddLast("šeši");        // įterpia naują mazgą gale
Console.WriteLine("Sąrašo narių kiekis: {0}", pirmas.Count);
```


Susieto sąrašo `LinkedList<T>` pavyzdžiai (4/18)

```
if (pirmas.Count > 0)
{
    Console.WriteLine("Pirmas susieto sąrašo elementas: {0}.",
                      pirmas.First.Value);
    Console.WriteLine("Paskutinis susieto sąrašo elementas:" +
                      " {0}.", pirmas.Last.Value);
    Console.WriteLine("    Susietas sąrašas pirmas");
    Console.WriteLine("    Reikšmės:");
    Spausdinti(pirmas);
}
else Console.WriteLine("Susietas sąrašas tuščias.");
Console.WriteLine();
```

```
Sąrašo narių kiekis:    3
Pirmas susieto sąrašo elementas: vienas.
Paskutinis susieto sąrašo elementas: šeši.
Susietas sąrašas pirmas
Reikšmės:
    vienas
    penki
    šeši
```

Press any key to continue . . .

Susieto sąrašo `LinkedList<T>` pavyzdžiai (5/18)

```
// RemoveFirst(), RemoveLast(), AddFirst(), AddLast()
LinkedListNode<string> mazgas = pirmas.Last;
pirmas.RemoveLast();           // išmeta paskutinį sąrašo mazgą
pirmas.AddFirst(mazgas);       // įterpia turimą mazgą pradžioje
Console.WriteLine("    Susietas sąrašas pirmas po pakeitimo");
Console.WriteLine("    Reikšmės:");
Spausdinti(pirmas);
mazgas = pirmas.First;
pirmas.RemoveFirst();          // išmeta pirmą sąrašo mazgą
pirmas.AddLast(mazgas);        // įterpia turimą mazgą sąrašo gale
Console.WriteLine("    Susietas sąrašas pirmas po antro pakeitimo");
Console.WriteLine("    Reikšmės:");
Spausdinti(pirmas);
```

```
Susietas sąrašas pirmas po pakeitimo
Reikšmės:
    šeši
    vienas
    penki
```

```
Susietas sąrašas pirmas po antro pakeitimo
Reikšmės:
    vienas
    penki
    šeši
```

Press any key to continue . . .

Susieto sąrašo `LinkedList<T>` pavyzdžiai (6/18)

```
LinkedList<Asmuo> antras = new LinkedList<Asmuo>();
Asmuo stud = new Asmuo("Jonas", 25);
Asmuo stud1 = new Asmuo("Petras", 26);
Asmuo stud2 = new Asmuo("Juozas", 38);
Asmuo stud3 = new Asmuo("Kazys", 25);
// AddFirst(), AddLast()
antras.AddFirst(stud);    // įterpia naują mazgą pradžioje
antras.AddLast(stud1);   // įterpia naują mazgą gale
antras.AddLast(stud2);   // įterpia naują mazgą gale
antras.AddLast(stud2);   // įterpia naują mazgą gale
Console.WriteLine("    Susietas sąrašas antras");
Console.WriteLine("Sąrašo narių kiekis:    {0}", antras.Count);
Console.WriteLine("    Reikšmės:");
Spausdinti(antras);
```

```
Susietas sąrašas antras
Sąrašo narių kiekis:    4
Reikšmės:
Jonas    25
Petras   26
Juozas   38
Juozas   38
```

Press any key to continue . . .

Susieto sąrašo `LinkedList<T>` pavyzdžiai (7/18)

```
LinkedListNode<Asmuo> mazgas2 = antras.Last;
LinkedListNode<Asmuo> mazgas3 =
    new LinkedListNode<Asmuo>(stud);
//antras.AddLast(mazgas2);    Klaida
antras.AddLast(mazgas3);
Console.WriteLine("    Susietas sąrašas antras");
Console.WriteLine("    Reikšmės:");
Spausdinti(antras);
```

```
Susietas sąrašas antras
Reikšmės:
Jonas 25
Petras 26
Juozas 38
Juozas 38
Jonas 25

Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (8/18)

```
// Sukuria ir inicializuoja susietą sąrašą
string[] skaic = { "pirmas", "antras", "trečias" };
LinkedList<string> trečias = new LinkedList<string>(skaic);
Console.WriteLine("    Susietas sąrašas trečias");
Console.WriteLine("Sąrašo narių kiekis:    {0}",
                  trečias.Count);
Console.WriteLine("    Reikšmės:");
Spausdinti(trečias);
```

```
Susietas sąrašas trečias
Sąrašo narių kiekis:    3
Reikšmės:
    pirmas
    antras
    trečias

Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (9/18)

```
// Sukuria ir inicializuoja aibę
HashSet<Asmuo> nauja = new HashSet<Asmuo>();
nauja.Add(stud);      // įterpia į aibę
nauja.Add(stud1);     // įterpia į aibę
nauja.Add(stud2);     // įterpia į aibę
nauja.Add(stud3);     // įterpia į aibę
Console.WriteLine("    Aibė nauja");
Console.WriteLine("Aibės nauja narių kiekis:    {0}",
                  nauja.Count);
Console.WriteLine("    Reikšmės:");
Spausdinti(nauja);    // aibės spausdinimo metodas
```

```
Aibė nauja
Aibės nauja narių kiekis:    4
Reikšmės:
Jonas      25
Petras     26
Juozas     38
Kazys      25
Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (10/18)

```
LinkedList<Asmuo> ketvirtas = new LinkedList<Asmuo>(nauja);
LinkedListNode<Asmuo> mazgas4 = new
    LinkedListNode<Asmuo>(stud);

ketvirtas.AddLast(mazgas4);
Console.WriteLine("    Susietas sąrašas ketvirtas");
Console.WriteLine("Sąrašo narių kiekis:    {0}",
    ketvirtas.Count);
Console.WriteLine("    Reikšmės:");
Spausdinti(ketvirtas);
```

```
Susietas sąrašas ketvirtas
Sąrašo narių kiekis:    5
Reikšmės:
Jonas  25
Petras 26
Juožas 38
Kazys  25
Jonas  25

Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (11/18)

```
// Find(), FindLast, AddAfter(), AddBefore(), Remove()
Asmuo stud4 = new Asmuo("Rita", 25);
LinkedListNode<Asmuo> mazgas5 = new LinkedListNode<Asmuo>(stud4);
LinkedListNode<Asmuo> mazgas6 = new LinkedListNode<Asmuo>(stud4);
mazgas2 = ketvirtas.FindLast(stud2); // ieško paskutinės reikšmės
if (mazgas2 != null)
{
    ketvirtas.AddAfter(mazgas2, stud); // įterpia reikšmę po surasto
    ketvirtas.AddAfter(mazgas2, mazgas5); // įterpia mazgą po surasto
    ketvirtas.AddBefore(mazgas2, stud1); // įterpia reikšmę prieš
    ketvirtas.AddBefore(mazgas2, mazgas6); // įterpia mazgą prieš
    Console.WriteLine("    Susietas sąrašas ketvirtas");
    Console.WriteLine("Sąrašo narių kiekis:    {0}", ketvirtas.Count);
    Console.WriteLine("    Reikšmės:");
    Spausdinti(ketvirtas);
}
else Console.WriteLine("mazgo mazgas2 susietame sąrašė ketvirtas nėra");
```


Susieto sąrašo `LinkedList<T>` pavyzdžiai (12/18)

```
// prieš tai buvusios skaidrės tęsinys
mazgas4 = ketvirtas.Find(stud4);           // ieško reikšmės
if (mazgas4 != null)
    Console.WriteLine("mazgo mazgas4 susietame sąraše " +
                      "surastas");
else Console.WriteLine("mazgo mazgas4 susietame sąraše " +
                      "ketvirtas nėra");
```

```
Susietas sąrašas ketvirtas
Sąrašo narių kiekis: 9
Reikšmės:
Jonas 25
Petras 26
Petras 26
Rita 25
Juozas 38
Rita 25
Jonas 25
Kazys 25
Jonas 25

mazgo mazgas4 susietame sąraše surastas
Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (13/18)

```
// Remove()
Asmuo stud5 = new Asmuo("Asta", 30);
Console.WriteLine("    Susietas sąrašas ketvirtas");
Console.WriteLine("    Reikšmės:");
Spausdinti(ketvirtas);
// išmeta pirmą surastą mazgą pagal jo reikšmę
ketvirtas.Remove(stud5); ketvirtas.Remove(mazgas2);
ketvirtas.Remove(mazgas2);    // išmeta nurodytą mazgą
Console.WriteLine("    Susietas sąrašas ketvirtas po " +
                  "pakeitimo");
Console.WriteLine("    Reikšmės:");
Spausdinti(ketvirtas);
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (14/18)

```
Susietas sąrašas ketvirtas
```

```
Reikšmės:
```

```
Jonas 25  
Petras 26  
Petras 26  
Rita 25  
Juozas 38  
Rita 25  
Jonas 25  
Kazys 25  
Jonas 25
```

```
Susietas sąrašas ketvirtas po pakeitimo
```

```
Reikšmės:
```

```
Jonas 25  
Petras 26  
Petras 26  
Rita 25  
Rita 25  
Jonas 25  
Kazys 25  
Jonas 25
```

```
Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (15/18)

```
//// Contains()  
if (ketvirtas.Contains(stud))  
    Console.WriteLine("rado reikšmę: {0} sąraše ketvirtas",  
                        stud);  
else Console.WriteLine("nerado reikšmės: {0} sąraše ketvirtas",  
                        stud);  
if (antras.Contains(stud4))  
    Console.WriteLine("rado reikšmę: {0} sąraše antras",  
                        stud4);  
else Console.WriteLine("nerado reikšmės: {0} sąraše antras",  
                        stud4);
```

```
rado reikšmę: Jonas 25 sąraše ketvirtas  
nerado reikšmės: Rita 25 sąraše antras  
Press any key to continue . . .
```

Susieto sąrašo `LinkedList<T>` pavyzdžiai (16/18)

```
// CopyTo()
Console.WriteLine("    Sąrašas antras:");
Spausdinti(antras);
Asmuo[] Masyvas = new Asmuo[antras.Count];
antras.CopyTo(Masyvas, 0);
Console.WriteLine("    Sąrašas antras perrašytas į masyvą:");
for (int i = 0; i < Masyvas.Length; i++)
    Console.WriteLine("    {0} ", Masyvas[i]);
Console.WriteLine();
```

Sąrašas antras:

```
Jonas 25
Petras 26
Juozas 38
Juozas 38
Jonas 25
```

Sąrašas antras perrašytas į masyvą:

```
Jonas 25
Petras 26
Juozas 38
Juozas 38
Jonas 25
```

Press any key to continue . . .

Susieto sąrašo `LinkedList<T>` pavyzdžiai (17/18)

```
// GetEnumerator() naudojamas skaityti, bet ne modifikuoti.
// Geriau ciklas foreach
var enumerator = antras.GetEnumerator();
Console.WriteLine("Enumeratoriaus pavyzdys");
Console.WriteLine("    Sąrašas antras");
while (enumerator.MoveNext())
{
    object item = enumerator.Current;
    Console.WriteLine("    {0}", item);
}
Console.WriteLine();
```

```
Enumeratoriaus pavyzdys
Sąrašas antras
Jonas 25
Petras 26
Juozas 38
Juozas 38
Jonas 25
```

Press any key to continue . . .

Susieto sąrašo `LinkedList<T>` pavyzdžiai (18/18)

```
var didz = ketvirtas.Max(elem => elem.amžius);  
Console.WriteLine("Max amžius: {0}.", didz);  
var didz2 = ketvirtas.Average(elem => elem.amžius);  
Console.WriteLine("Amžiaus vidurkis: {0, 5:f}.", didz2);
```

```
Max amžius: 38.  
Amžiaus vidurkis: 26,67.  
Press any key to continue . . .
```



Klausimai?