

İsim: NATHANAELLE BOPTI NGAH BONG  
Öğrenci Numarası : 24080410150  
Bilgisayar Mühendisliği

## İKİ YÖNLÜ BAĞLI LİSTE

```
using System.Security.Cryptography.X509Certificates;

public class Node
{
    public int Veri;
    public Node Sonraki;
    public Node Onceki;

    public Node(int veri)
    {
        Veri = veri;
        Sonraki = null;
        Onceki = null;
    }
}

public class CiftYonlu_Bagli_Liste
{
    private Node head;
    private Node tail;
    public int boyut;

    public CiftYonlu_Bagli_Liste()
    {
        head = null;
        tail = null;
        boyut = 0;
    }

    // Başa Ekleme
    public void Başa_Ekle(int yeni_değeri)
    {
        Node yeniNode = new Node(yeni_değeri);
        if (head == null)
        {
            head = yeniNode;
            tail = yeniNode;
        }
        else
        {
            yeniNode.Sonraki = head;
            head.Onceki = yeniNode;
            head = yeniNode;
        }
        boyut++;

        Console.WriteLine($"{yeni_değeri} başa eklendi");
    }
}
```

```

// Sona ekleme

public void Sona_Ekle(int yeni_değeri)
{
    Node yeniNode = new Node(yeni_değeri);
    if (tail == null)
    {
        head = yeniNode;
        tail = yeniNode;
    }
    else
    {
        yeniNode.Onceki = tail;
        tail.Sonraki = yeniNode;
        tail = yeniNode;
    }
    boyut++;
    Console.WriteLine($"{yeni_değeri} Sona eklendi ");
}

// Araya herhangi bir veriden sonra ekleme

public void veridenSonraEkleme(int hedef_veri, int yeni_değeri)
{
    Node mevcut = head;
    while (mevcut != null && mevcut.Veri != hedef_veri)
    {
        mevcut = mevcut.Sonraki;
    }
    if (mevcut == null)
    {
        Console.WriteLine($"Hata: {hedef_veri} listede bulunamadı. Ekleme yapılmadı. ");
        return;
    }
    if (mevcut == tail)
    {
        Sona_Ekle(yeni_değeri);
    }
    else
    {
        Node yeniNode = new Node(yeni_değeri);
        yeniNode.Sonraki = mevcut.Sonraki;
        yeniNode.Onceki = mevcut;
        mevcut.Sonraki.Onceki = yeniNode;
        mevcut.Sonraki = yeniNode;
        boyut++;

        Console.WriteLine($"{yeni_değeri} , {hedef_veri} verisinden sonra eklendi");
    }
}

// Araya herhangi Bir Veriden önce Ekleme
public void VeridenOnceEkleme(int hedef_veri, int yeni_değeri)
{
    Node mevcut = head;

```

```

        while (mevcut != null && mevcut.Veri != hedef_veri)
        {
            mevcut = mevcut.Sonraki;
        }

        if (mevcut == null)
        {
            Console.WriteLine($"Hata : {hedef_veri} listede bulunamadı. Ekleme yapılmadı");
            return;
        }

        if (mevcut == head)
        {
            Başa_Ekle(yeni_değeri);
        }
        else
        {
            Node yeniNode = new Node(yeni_değeri);
            yeniNode.Onceki = mevcut.Onceki;
            yeniNode.Sonraki = mevcut;
            mevcut.Onceki.Sonraki = yeniNode;
            mevcut.Onceki = yeniNode;
            boyut++;

            Console.WriteLine($"{yeni_değeri}, {hedef_veri} verisinden önce eklendi.");
        }
    }

    // Baştan silme
    public void Baştan_silme()
    {
        if (head == null)
        {
            Console.WriteLine("Hata : Liste zaten boş .");
        }
        int silinenVeri = head.Veri;
        if (head == tail)
        {
            head = null;
            tail = null;
        }
        else
        {
            head = head.Sonraki;
            head.Onceki = null;
        }
        boyut--;
        Console.WriteLine($"{silinenVeri} baştan silindi");
    }

    // Sondan silme
    public void SondanSil()
    {
        if (tail == null)

```

```

    {
        Console.WriteLine("Hata : Liste zaten boş . ");
    }
    int silinenVeri = tail.Veri;
    if (head == tail)
    {
        head = null;
        tail = null;
    }
    else
    {
        tail = tail.Onceki;
        tail.Sonraki = null;
    }
    boyut--;
    Console.WriteLine($"{silinenVeri} sondan silindi .");
}

// Aradan arayarak silme
public void AradanSil(int hedefVeri)
{
    Node mevcut = head;
    while (mevcut != null && mevcut.Veri != hedefVeri)
    {
        mevcut = mevcut.Sonraki;
    }
    if (mevcut == null)
    {
        Console.WriteLine($"Hata : {hedefVeri} listede bulunamadı. Silme yapılmadı");
    }
    if (mevcut == head)
    {
        Baştan_silme();
    }
    else if (mevcut == tail)
    {
        SondanSil();
    }
    else
    {
        mevcut.Onceki.Sonraki = mevcut.Sonraki;
        mevcut.Sonraki.Onceki = mevcut.Onceki;

        boyut--;
        Console.WriteLine($"{hedefVeri} aradan silindi");
    }
}

// Arama
public bool Ara(int hedefVeri)
{
    Node mevcut = head;
    int pozisyon = 0;
    while (mevcut != null)

```

```

        {
            if(mevcut.Verii== hedefVeri)
            {
                Console.WriteLine($"{hedefVeri} verisi listenin {pozisyon}.
pozisyonunda bulundu");
                return true;
            }
            mevcut = mevcut.Sonraki;
            pozisyon++;
        }
        Console.WriteLine($"{hedefVeri} verisi listede bulunamadı");
        return false;
    }

    // Listeleme
    public void Listele ()
    {
        if (head== null)
        {
            Console.WriteLine("Liste boş");
        }
        Console.WriteLine($"Liste içeriği (Boyut: {boyut}) ");
        Node mevcut = head;
        while (mevcut!= null)
        {
            Console.WriteLine($"{mevcut.Verii}");
            if(mevcut.Sonraki!= null)
            {
                Console.WriteLine("<->");
            }
            mevcut = mevcut.Sonraki;
        }
        Console.WriteLine("<-head\n");
    }

    // Tümünü Silme
    public void TumunuSil()
    {
        head = null;
        tail = null;
        boyut = 0;
        Console.WriteLine("Tüm liste temizlendi. ");
    }

    // Tüm Linked Listi Bir Diziye atma
    public int[] DiziyeAt()
    {
        if(boyut==0)
        {
            return new int[0];
        }
        int[] dizi = new int[boyut];
        Node mevcut = head;
        int index = 0;

```

```

        while (mevcut != null)
        {
            dizi[index++] = mevcut.Veri;
            mevcut = mevcut.Sonraki;
        }
        Console.WriteLine("Liste başarılı bir şekilde diziye aktarıldı");
        return dizi;
    }
}

public class Program
{
    public static void Main(string[] args)
    {
        CiftYonlu_Bagli_Liste liste = new CiftYonlu_Bagli_Liste();
        liste.Sona_Ekle(10);
        liste.Sona_Ekle(35);
        liste.Basa_Ekle(6);
        liste.Sona_Ekle(43);

        liste.Listele();
        liste.veridenSonraEkleme(10, 20);
        liste.VeridenOnceEkleme(40, 35);
        liste.Listele();

        liste.Bastan_silme();
        liste.SondanSil();
        liste.AradanSil(6);
        liste.Ara(20);
        int[] dizisonucu = liste.DiziyeAt();
        liste.TumunuSil();
    }
}

```

```
10 Sona eklendi
35 Sona eklendi
6 basa eklendi
43 Sona eklendi
Liste icerigi (Boyut: 4) 6<->
10<->
35<->
43<-head

20 , 10 verisinden sonra eklendi
Hata : 40 listede bulunamadi. Ekleme yapilmadi
Liste icerigi (Boyut: 5) 6<->
10<->
20<->
35<->
43<-head

6 bastan silindi
43 sondan silindi .
Hata : 6 listede bulunamadi. Silme yapilmadi
}
```