

Tekli Ve Çiftli Linked List ile Stack İşlemleri

```
static public int sp = -1;
static public int[] stack = new int[100];

static public void push(int data)
{
    sp++;
    stack[sp] = data;
}

static public int pop()
{
    int data = stack[sp];
    sp--;
    return data;
}

static public int read()
{
    int data = stack[sp];
    return data;
}

static Tekli sp_ = null;
static Tekli head_ = null;
static public void push_(int data)
{
    Tekli q = new Tekli();
    q.data = data;
    q.next = null;
    if(sp_ == null)
    {
        sp_ = q;
        head_ = q;
    }
    else
    {
        sp_.next = q;
        sp_ = q;
    }
}
```

```
static public int pop_()
{
    int data = sp_.data;
    Tekli temp = head_;
    while(temp.next != sp_)
    {
        temp = temp.next;
    }
    temp.next = null;
    sp_ = temp;

    temp = sp_;
    while(temp.next != sp_)
    {
        temp = sp_;
    }
    temp.next = sp_.next;

    return data;
}
```

```
static Ciftli sp__ = null;
static public void push__(int data)
{
    Ciftli q = new Ciftli();
    q.data = data;
    q.next = null;
    q.prev = sp__;
    if(sp__ == null)
    {
        sp__ = q;
    }
    else
    {
        sp__.next = q;
        sp__ = q;
    }
}
```

```
static public int pop__()
{
    int data = sp__.data;
    sp__ = sp__.prev;
    return data;
}
```

```
static int count()
{
    int adet = 0;
    Ciftli w = sp__;
    while (w != null)
    {
        w = w.prev;
        adet++;
    }
    return adet;
}

static int Count_(Ciftli q)
{
    if (q == null) return 0;
    return 1 + Count_(q.prev);
}

class Tekli
{
    public int data;
    public Tekli next;
}

class Ciftli
{
    public int data;
    public Ciftli next;
    public Ciftli prev;
}
```