

# Delegates and events

```
delegate void BossUpgrade(string str);
delegate void BossAttack(int bossPower);
class Boss
{
    public event BossUpgrade Upgraded;
    public event BossAttack Attacked;

    int _power = 1;

    public void Upgrade()
    {
        _power++;
        Upgraded?.Invoke(_power.ToString());
    }

    public void TurnOn()
    {
        Attacked?.Invoke(_power);
    }
}

class Worker
{
    int _voltage;
    string _name;
    bool _isLive = true;
    public Worker(string name, int voltage)
    {
        this._voltage = voltage;
        this._name = name;
    }

    public void TakeVoltage(int _bossPower)
    {
        if (_isLive && _voltage < _bossPower)
        {
            Console.WriteLine($"{_name} Смерть");
            _isLive = false;
        }
        else if (_isLive)
        {
            Console.WriteLine($"{_name} норм");
        }
    }
}

class Task01
{
    public static void Main()
    {
        Boss boss = new Boss();
        Worker worker = new Worker("Дима", 1);
        Worker worker2 = new Worker("Никита", 2);
        boss.Attacked += worker.TakeVoltage;
        boss.Attacked += worker2.TakeVoltage;
        boss.Upgraded += DisplayMessage;

        boss.TurnOn();

        boss.Upgrade();
        boss.TurnOn();

        boss.Upgrade();
        boss.TurnOn();
    }
}
```

```

void DisplayMessage(string message) => Console.WriteLine($"Теперь босс имеет напряжение {message}");

string str1 = "Hello world!";

void DelE(string st1, int ind, Action<string, int> op) => op(st1, ind);
Predicate<string> IsFirstH = (string s1) => s1.IndexOf('H') == 0;
Func<string, string, int, string> Insert = (string s1, string s2, int i) => InsEl(s1, s2, i);

DelE(str1, 10, DelElems);
Console.WriteLine(str1);

Console.WriteLine(IsFirstH(str1));
Console.WriteLine(Insert(str1, "How", 4));

void DelElems(string st1, int ind) => str1 = st1.Remove(ind);
string InsEl(string st1, string st2, int ind) => str1.Insert(ind, st2);
void ReplaceS(string st1, char c1, char c2) => str1.Replace(c1, c2);
void Substr(string str1, int ind) => str1.Substring(ind);
void IndexOfC(string str1, char c1) => str1.IndexOf(c1);
}
}

```

## Контрольная работа 2

```

using Microsoft.VisualBasic;
using System;
using System.Net.Sockets;

delegate void DrSayUP();

class Dressirovchik
{
    public event DrSayUP SaidUp;

    public void UP()
    {
        Console.WriteLine("Дрессировщик: UP!");
        SaidUp?.Invoke();
    }
}

class Horse
{
    string _name;
    public Horse(string name)
    {
        _name = name;
    }

    public void DoUp()
    {
        Console.WriteLine($"{_name} *встает на дыбы*");
    }
}

class SuperLinkedList<T> : LinkedList<T>
{
    public new void Remove(T item)
    {
        if (!Contains(item))
            throw new Exception($"Элемента {item} нет в {this.ToString()}");
        Console.WriteLine($"Удаление {item} из {this.ToString()}");
    }
}

```

```

        base.Remove(item);
    }
}

public class Program
{
    public static void Main(string[] args)
    {
        Dressirovchik dressirovchik = new Dressirovchik();

        Horse horse = new Horse("Стрела");
        Horse horse1 = new Horse("Победа");
        Horse horse2 = new Horse("Непослушная лошадь");

        dressirovchik.SaidUp += horse.DoUp;
        dressirovchik.SaidUp += horse1.DoUp;

        dressirovchik.UP();
        Console.WriteLine();

        try
        {
            SuperLinkedList<string> sll = new SuperLinkedList<string>();
            sll.AddLast("hello");
            sll.AddFirst("hellow");
            sll.AddFirst("1234567");
            sll.AddFirst("123456");
            sll.AddFirst("12345");
            sll.AddFirst("1234");

            Console.WriteLine($"B {sll.ToString()} находится: ");
            foreach (string i in sll)
            {
                Console.WriteLine(i);
            }
            Console.WriteLine();

            sll.Remove("1234567");

            Console.WriteLine($"B {sll.ToString()} находится: ");
            foreach (string i in sll)
            {
                Console.WriteLine(i);
            }
            Console.WriteLine();

            /*
                sll.Remove("12345678");*/

            Console.WriteLine($"Элементы {sll.ToString()} длины строк которых < 6: ");
            var selectedSLL = from p in sll
                              where p.Length < 6
                              select p;

            foreach (string si in selectedSLL)
            {
                Console.WriteLine(si);
            }
        }
        catch (Exception e)
        {
            Console.WriteLine(e.Message);
        }
    }
}

```