



**ВЛОЖЕН УСЛОВЕН ОПЕРАТОР.**

# 1. ВЛОЖЕН УСЛОВЕН ОПЕРАТОР

Понякога се налага в частта `if` или в частта `else`, или и в двете да се провери друго условие, т.е. На това място ще се използва нов условен оператор. В такъв случай казваме, че имаме вложен условен оператор.



## Зад:1

```
static void Main(string[] args)
{
    int a, b, c;
    a = -3; b = 3; c = 0;
    if(a>0)
    { if (b>0)
      { c = 20; } //a>0 b>0
      else
      { c = -20; } //a>0 b<=0
    }
    Console.WriteLine(c);
}
```

C:\Windows\system32\cmd.exe

Press any key to continue . . .

Зад2:

```
static void Main(string[] args)
{
    int a, b, c;
    a = -3; b = 3; c = 0;
    if (a > 0)
        if (b > 0)
            c = 20; //a>0 b>0
        else
            c = -20; //a>0 b<=0
    Console.WriteLine(c);
}
```

C:\Windows\system32\cmd.exe

Press any key to continue . . .

В зад:2 къдрави скоби няма, защото когато се използва един оператор къдравите скоби могат да се пропуснат. В C# няколко оператора когато се поставят в къдрави скоби образуват блок от оператори. Блокът от оператори се разглежда като един оператор. If часта и else часта образуват един оператор, а не два. Правило: всяко else се отнася за най-близко стоящата част if.



Зад3:

```
static void Main(string[] args)
{
    int a, b, c;
    a = -3; b = 3; c = 0;
    if (a > 0)
    {
        if (b > 0)
            c = 20; //a>0 b>0
    }
    else
        c = -20; //a<=0
    Console.WriteLine(c);
}
```

C:\Windows\system32\cmd.exe

-20

Press any key to continue . . . \_

## 2. ЕЛИМИНИРАНЕ НА ВЛОЖЕН УСЛОВЕН ОПЕРАТОР.

Ако имаме два оператора if без частта else, тогава втория вложен оператор if може да бъде елиминиран като двете условия се съберат с операцията конюнкция.

Зад:4

```
static void Main(string[] args)
{
    int a, b, c;
    a = -3; b = 3; c = 0;
    if (a > 0)
    { if (b > 0)
      { c = 20; }
    }
    Console.WriteLine(c);
}
```

Зад5: еквивалентна на зад:4

```
static void Main(string[] args)
{
    int a, b, c;
    a = -3; b = 3; c = 0;
    if (a > 0 && b > 0)
    { c = 20; }
    Console.WriteLine(c);
}
```

C:\Windows\system32\cmd.exe

Press any key to continue . . . \_



## 3. РАБОТА С КОМПЮТЪР

### Зад:6

Напишете програма с графичен интерфейс, която намира периметъра и лицето на правоъгълник като прави проверка за коректността на входните данни.



Form1



Въведи страна a:

Въведи страна b:

Пресметни

Периметър: 0

Лице:

0

## Код на бутона:

```
{
```

```
    int a = int.Parse(textBox1.Text);
```

```
    int b = int.Parse(textBox2.Text);
```

```
    if (a>0 && b>0)
```

```
    { int p = 2 * (a + b);
```

```
    label5.Text = String.Format("{0}", p);
```

```
    int s = a * b;
```

```
    label6.Text = String.Format("{0}", s);
```

```
    }
```

```
    if (a>0 && b<=0)
```

```
    { label5.Text = String.Format("Некоректно въведена страна b");
```

```
    label6.Text = String.Format("Некоректно въведена страна b");
```

```
    }
```

```
    if (a<=0)
```

```
    { label5.Text = String.Format("Некоректно въведена страна a");
```

```
    label6.Text = String.Format("Некоректно въведена страна a");
```

```
    }
```

```
}
```

