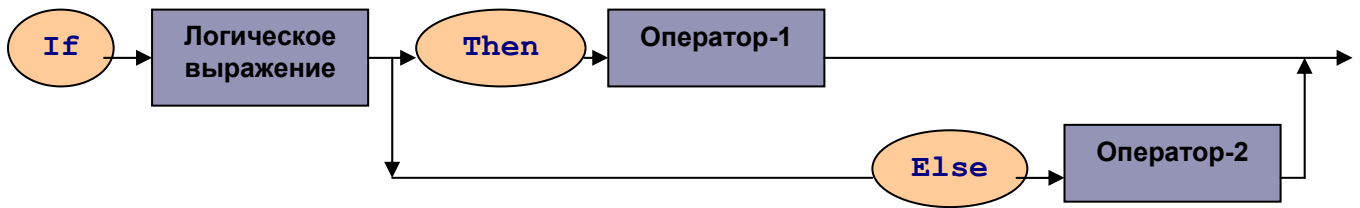
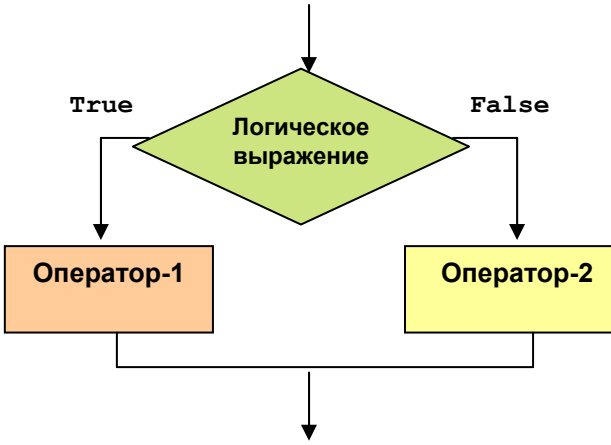
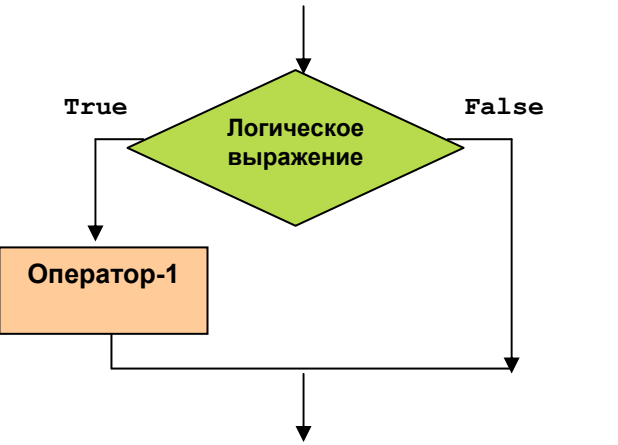


## Оператор условия If



Последовательность выполнения оператора.

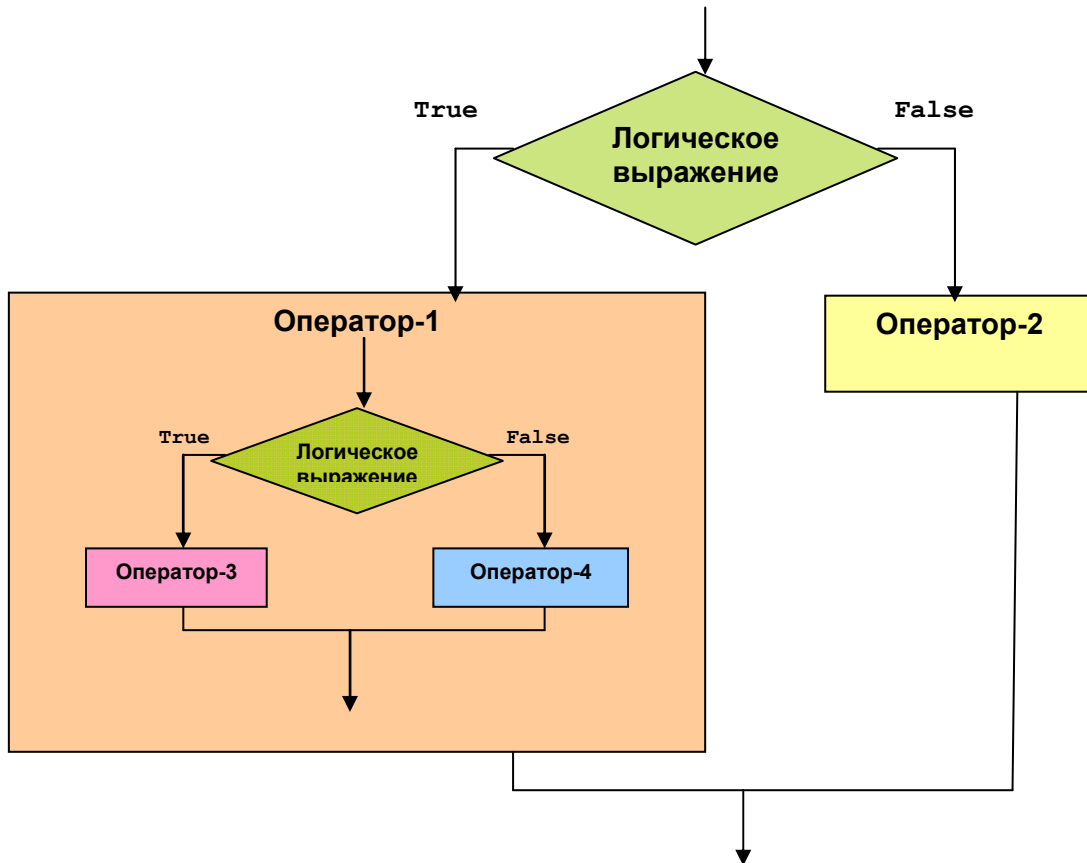
- Вычисляется логическое выражение (булевый тип: True, False),
- Если значение логического выражения есть True (истина), выполняется <Оператор-1>, указанный после Then,
- Если значение логического выражения есть False (ложь), выполняется <Оператор-2>, указанный после Else.

Полная форма	Неполная форма
	
<pre><b>If</b> &lt;Логическое выражение&gt; <b>Then</b> &lt;Оператор-1&gt; <b>Else</b> &lt;Оператор-2&gt;;</pre>	<pre><b>If</b> &lt;Логическое выражение&gt; <b>Then</b> &lt;Оператор-1&gt;;</pre>
<p>Если логическое выражение принимает значение <i>True</i>, то выполняется <i>Оператор-1</i>, в противном случае – <i>Оператор-2</i>.</p>	<p>Если логическое выражение принимает значение <i>True</i>, то выполняется <i>Оператор-1</i>, в противном случае – выполняется оператор идущий за условным оператором..</p>
<b>Пример</b>	
<pre><b>If</b> a = b <b>Then</b> a:=5 <b>Else</b> b:= a+10;</pre>	<pre><b>If</b> a = b <b>Then</b> a:=5;</pre>

**Примечание**

Теоретически можно сделать, чтобы *Оператор-1* отсутствовал, но это неграмотное решение только осложняет понимание алгоритма и запутывает его.

В качестве *Оператор-1* или *Оператор-2* может стоять оператор ветвления:



```

If <Логическое выражение>
  Then {Оператор-1}
    If <Логическое выражение>
      Then <Оператор-3>
      Else <Оператор-4>
    Else <Оператор-2>;
    
```

**Пример**

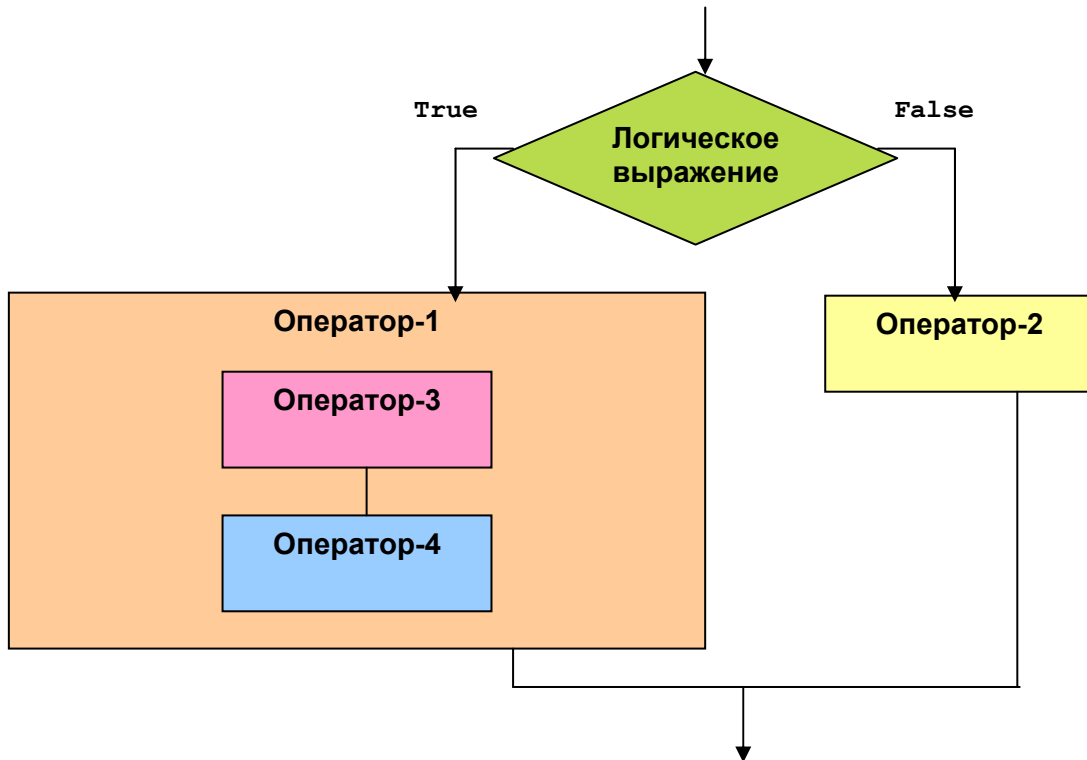
```

If a = b
  Then
    If a > 5
      Then a := a - 5
      Else a := a + 5
    Else b := a + 10;
    
```

**Примечание**

При записи операторов для наглядности целесообразно использовать сдвиг.

В качестве *Оператор-1* или *Оператор-2* может стоять несколько операторов:



Использование составного оператора (операторных скобок): **begin-end**

```

If <Логическое выражение>
  Then {Оператор-1}
    begin
      <Оператор-3>;
      <Оператор-4>
    end
  Else <Оператор-2>;
  
```

Пример

```

If a = b
  Then
    begin
      a:=a-5;
      b:=a+5
    end
  Else b:= a+10;
  
```

Чтобы понять, как работает программа, нужно определить, к какому из условных операторов **if** относится часть **else**.

Для этого используется правило: «**else** относится к ближайшему **if**»

