**Модуль bisect**

Модуль **bisect** - обеспечивает поддержку списка в отсортированном порядке с помощью алгоритма деления пополам.

Набор функций:

**bisect.insort(list, elem)**, он же **bisect.insort\_right(list, elem)** - вставка элемента в отсортированный [список](https://pythonworld.ru/tipy-dannyx-v-python/spiski-list-funkcii-i-metody-spiskov.html), при этом elem располагается как можно правее (все элементы, равные ему, остаются слева).

**bisect.insort\_left(list, elem)** - вставка элемента в отсортированный список, при этом elem располагается как можно левее (все элементы, равные ему, остаются справа).

**bisect.bisect(list, elem)**, он же **bisect.bisect\_right(list, elem)** - поиск места для вставки элемента в отсортированный список, таким образом, чтобы elem располагался как можно правее.

**bisect.bisect\_left(list, elem)** - поиск места для вставки элемента в отсортированный список, таким образом, чтобы elem располагался как можно левее.

Для полного счастья не хватает только функции для проверки наличия элемента в отсортированном списке. К счастью, это легко решаемо.

>>>

**>>> from** **bisect** **import** bisect\_left

**>>> def** contains(l, elem):

**...**  index = bisect\_left(l, elem)

**...**  **if** index < len(l):

**...**  **return** l[index] == elem

**...**  **return** False

**...**

**>>>** contains(list(range(1000)), -10)

False

**>>>** testlist = (1, 2, 3, 6, 8, 10, 15)

**>>>** contains(testlist, 10)

True

**>>>** contains(testlist, 0)

False

**>>>** contains(testlist, 20)

False