

```
from tkinter import *
import tkinter.messagebox as msg
import timeit
#import unittest

# К сожалению, с unittest не получилось разобраться.
#class TestSorting(unittest.TestCase):
#    def testSort(self):
#        self.assertEqual(action('5,4,3,2,1'), [1,2,3,4,5])

'''
Функция считывания данных из окна ввода и сортировки введённых чисел.
Функция запускается при нажатии на кнопку <Старт>.
Функция ничего не получает и ничего не возвращает, работает непосредственно
с элементами окна.
'''
def action():
    fail = FALSE
    # Считываем введённую строку
    source = entry.get()
    # Преобразуем строку в список
    sList = source.split(',')
    # Преобразуем тип в int
    for i in range(len(sList)):
        try:
            sList[i] = int(sList[i])
        except:
            fail = TRUE
    # Если преобразование прошло успешно, то переходим к сортировке
    # Иначе выводим сообщение, что произошла ошибка преобразования типов
    if not fail:
        method = variable.get()
        start_time = timeit.default_timer()
        # Выполняется встроенная сортировка списка
        if method == 'Встроенная':
            sList.sort()
            resV.config(text=str(sList))
        # Выполняется сортировка пузырьком
        else:
            for i in range(len(sList) - 1):
                for j in range(len(sList) - 1):
                    if sList[j] > sList[j + 1]:
                        sList[j], sList[j + 1] = sList[j + 1], sList[j]
            resV.config(text=str(sList))
            durationV.config(text=str(timeit.default_timer() - start_time))
    else:
        msg.showerror('Ошибка', 'Ошибка преобразования типов. Повторите
        ввод.')
    # return sList

# Главная функция
# Построение внешнего вида окна программы
wind = Tk()
```

```
wind.title('Выберите способ сортировки')
wind.geometry('500x130')
begin = Label(wind, text='Исходная последовательность:')
entry = Entry(wind)
method = Label(wind, text='Метод сортировки:')
menu = ['Встроенная', 'Пузырьком']
variable = StringVar(wind)
variable.set(menu[0])
dropdown = OptionMenu(wind, variable, *menu)
resN = Label(wind, text='Результат: ')
resV = Label(wind)
durationN = Label(wind, text='Время выполнения:')
durationV = Label(wind)
btn = Button(wind, text='Старт', command=action)
# Для размещения элементов интерфейса используется метод GRID.
begin.grid(row=0, column=0)
method.grid(row=0, column=1)
durationN.grid(row=0, column=2)
entry.grid(row=1, column=0)
resN.grid(row=2, column=0)
resV.grid(row=3, column=0)
entry.focus()
dropdown.grid(row=1, column=1)
durationV.grid(row=1, column=2)
btn.grid(row=3, column=2)

#if __name__ == '__main__':
#    unittest.main()

wind.mainloop()
```