

Информатика и математическое моделирование

Контактное лицо: Медведев Илья Дмитриевич

e-mail для переписки: ilyafregit@gmail.com

телефон для срочной связи: 89090863515

Преобразование звука в нотный текст

Медведев Илья Дмитриевич

11

Муниципальное автономное образовательное учреждение «Лицей №77 города Челябинска», Челябинск, Россия

Научный руководитель: Разумовская Галина Викторовна, Муниципальное автономное общеобразовательное учреждение «Лицей №77 города Челябинска», учитель информатики

[I.Medvedev](#)

Converting audio-format to notation

ПРЕОБРАЗОВАНИЕ ЗВУКА В НОТЫ

Автор:

Медведев Илья Дмитриевич, «11А» класс,
МАОУ «Лицей № 77 г. Челябинска»

Руководитель:

Разумовская Галина Викторовна,
учитель информатики,
МАОУ «Лицей № 77 г. Челябинска»

Челябинск – 2021

Почему подобные программы могут проигрывать написанную музыку, но не имеют возможности самим записывать нотную запись музыки, проигрываемой человеком вживую?



SmartScore 64
MIDI Edition
Scan & Play. Turn sheet music and PDF files into playable MIDI files for just \$49! Limitations: Outputs to MIDI only. No chord symbol, guitar fretboard, text or lyric recognition.
\$49

RUR **USD** EUR GBP ▼

49
1 USD = 75.7023 RUR

31.03.2021 📅

ЦБ РФ ▼

RUR USD EUR GBP ▼

3 709.4122
1 RUR = 0.0132 USD



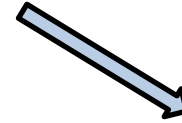
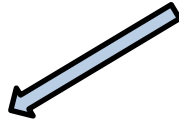
Данные за 2021-03-31 19:35:54 GMT+03:00

Цель работы – написание программы для преобразования звуковой дорожки в нотный текст на языке программирования *python*.



Guido van Rossum,
разработчик языка
python

Актуальность



Трудность

нахождения аналогов

Простота

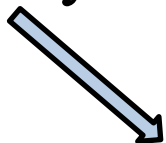
пользования

Задачи

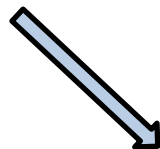
- Изучение аспектов языка программирования python
- Изучить алгоритмы цифровой обработки звук
- Реализовать полученный алгоритм, а также способ вывода результата работы проекта

Теоретическая часть

Получить звуковую дорожку



«Раздробить» ее на много маленьких кусочков — *фреймов*, обработать каждый кусочек отдельно, зафиксировав его частоту



Присвоить конкретное значение ноты полученной частоты

Преобразования Фурье



Jean-Baptiste Joseph Fourier, известный французский математик и физик (1768–1830). Основатель Метода преобразований Фурье.

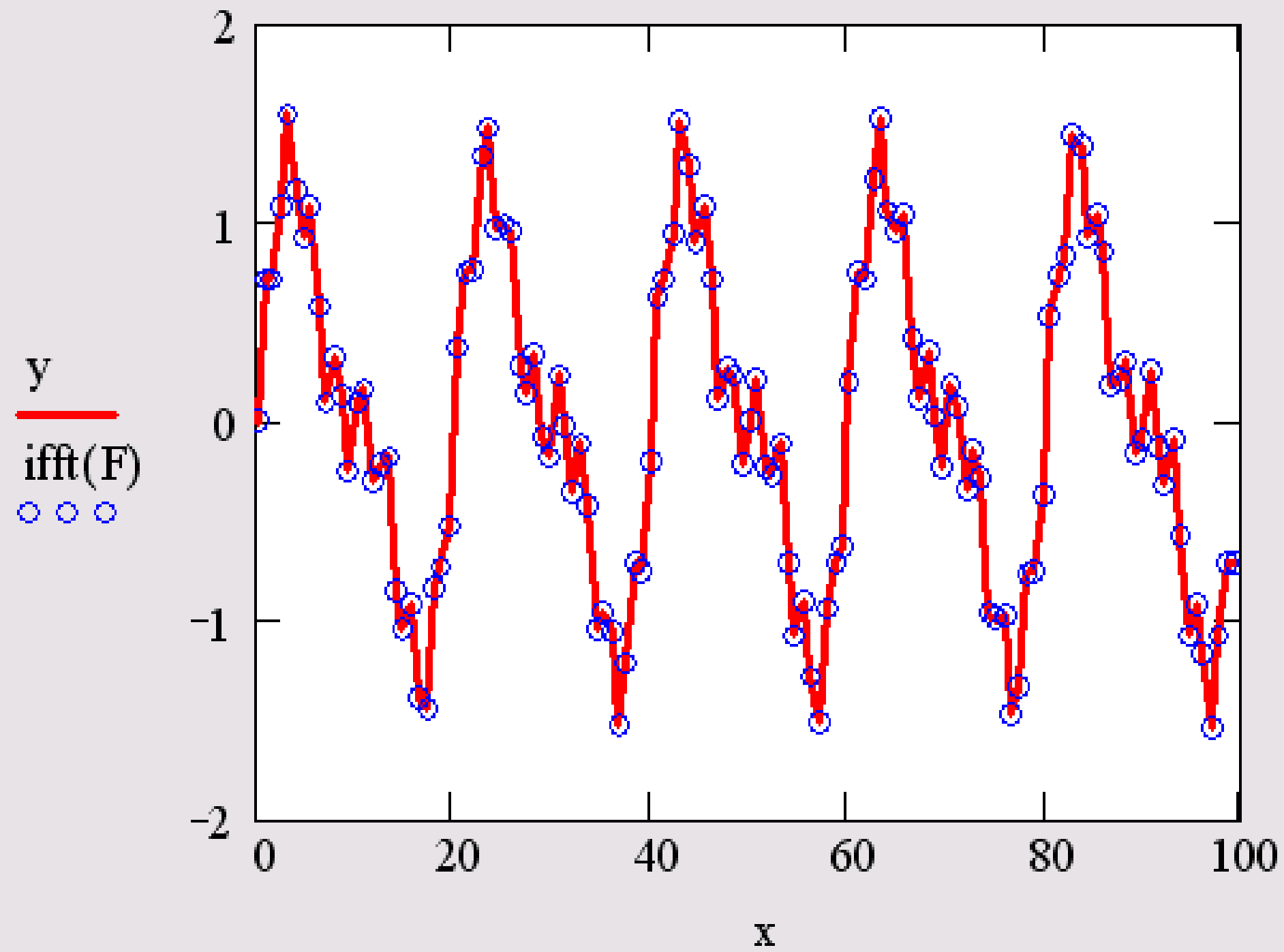
Алгоритм Кули-Тьюки (1965)

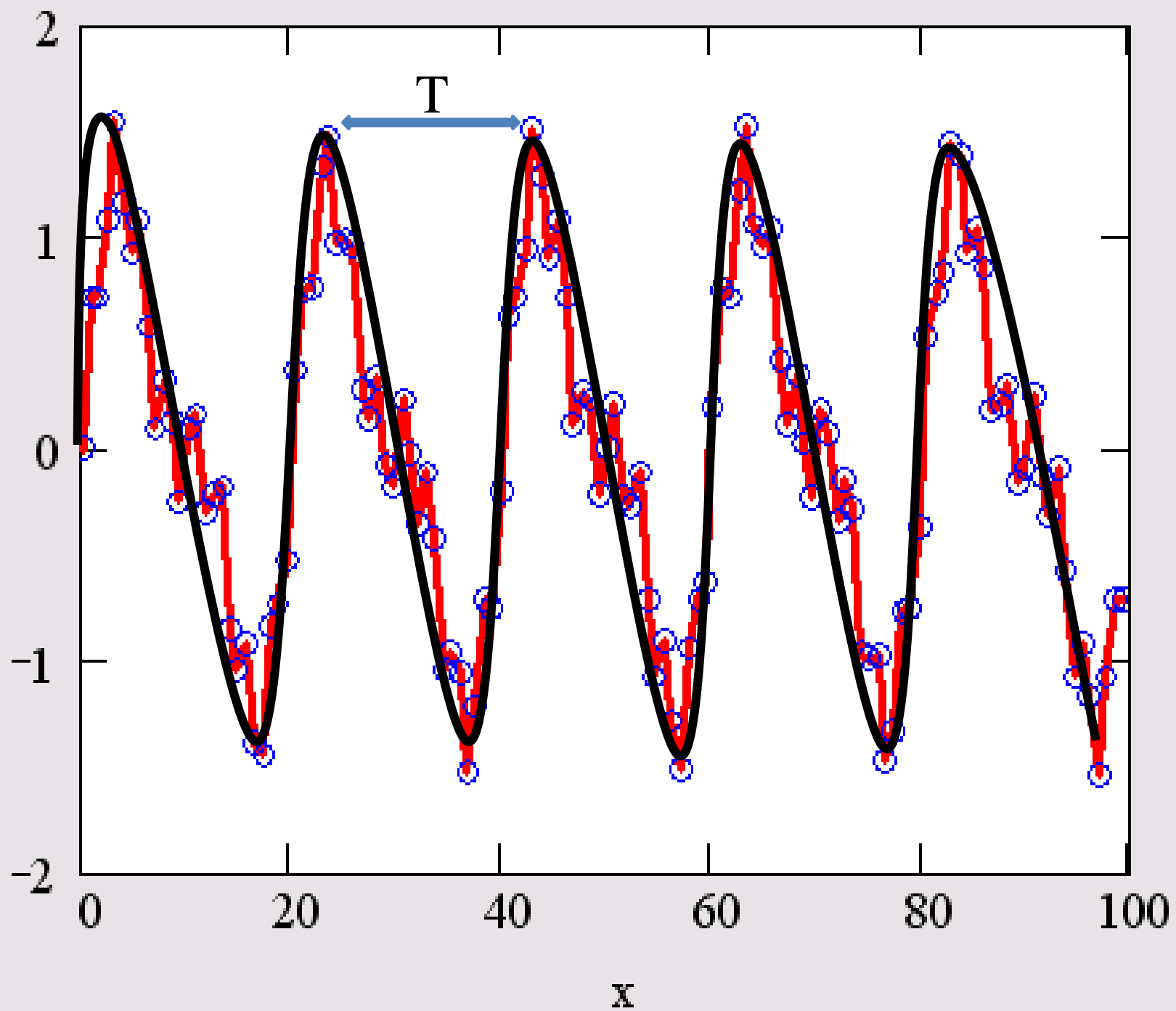


James Cooley
(1926 – 2016)



John Tukey
(1915 – 2000)





MIDI number	Note name	Keyboard	Frequency Hz	Period ms
21	A0		27.500	36.36
23	B0		30.868	29.135
24	C1		32.703	30.58
26	D1		36.708	34.648
28	E1		41.203	38.891
29	F1		43.654	22.91
31	G1		48.999	46.249
32	A1		55.000	51.913
33	B1		61.735	58.270
35	C2		65.406	15.29
36	D2		73.416	69.296
38	E2		82.407	77.782
40	F2		87.307	11.45
41	G2		97.999	92.499
43	A2		110.00	103.83
45	B2		123.47	116.54
47	C3		130.81	7.645
48	D3		146.83	138.59
50	E3		164.81	155.56
52	F3		174.61	5.727
53	G3		196.00	185.00
55	A3		220.00	207.65
57	B3		246.94	233.08
59	C4		261.63	3.822
60	D4		293.67	277.18
62	E4		329.63	311.13
64	F4		349.23	2.863
65	G4		392.00	369.99
67	A4		440.00	4.545
69	B4		493.88	466.16
71	C5		523.25	1.910
72	D5		587.33	554.37
74	E5		659.26	622.25
76	F5		698.46	1.432
77	G5		783.99	739.99
79	A5		880.00	830.61
81	B5		987.77	932.33
83	C6		1046.5	0.9556
84	D6		1174.7	1108.7
86	E6		1318.5	1244.5
88	F6		1396.9	0.7159
89	G6		1568.0	1480.0
91	A6		1760.0	1661.2
93	B6		1975.5	1864.7
95	C7		2093.0	0.4778

$$n_0 = \log_2 \frac{f_1}{f_2}$$

$$f_n = 2^{\frac{n}{12}} * 440$$

$$n = 12 * \log_2 \frac{f_n}{440}$$

$$n = 12 * \log_2 \frac{f_n}{440} + 69$$

```
def freq_to_note(frequency): # Функция сопоставляет значение частоты звука с нотой
```

```
    midi_number = (69 + 12*np.log2(frequency/440.0)) # формула выведена на основе данных сайта:
```

```
    n = midi_number.astype(np.int)
```

```
    #https://newt.phys.unsw.edu.au/jw/notes.html
```

```
    notes = 'C C# D D# E F F# G G# A A# B'.split()
```

```
    return notes[n % 12] # + str(round(n/12 - 1)) <- следует добавить в return при необходимости определения октавы.
```

```
def notation_to_out(arr): # Функция, получая на ввод массив нот, создает word-файл,
                          # вставляет в него последовательно картинки нот,
                          # сопоставляя их с каждым элементом массива.

    document = Document()

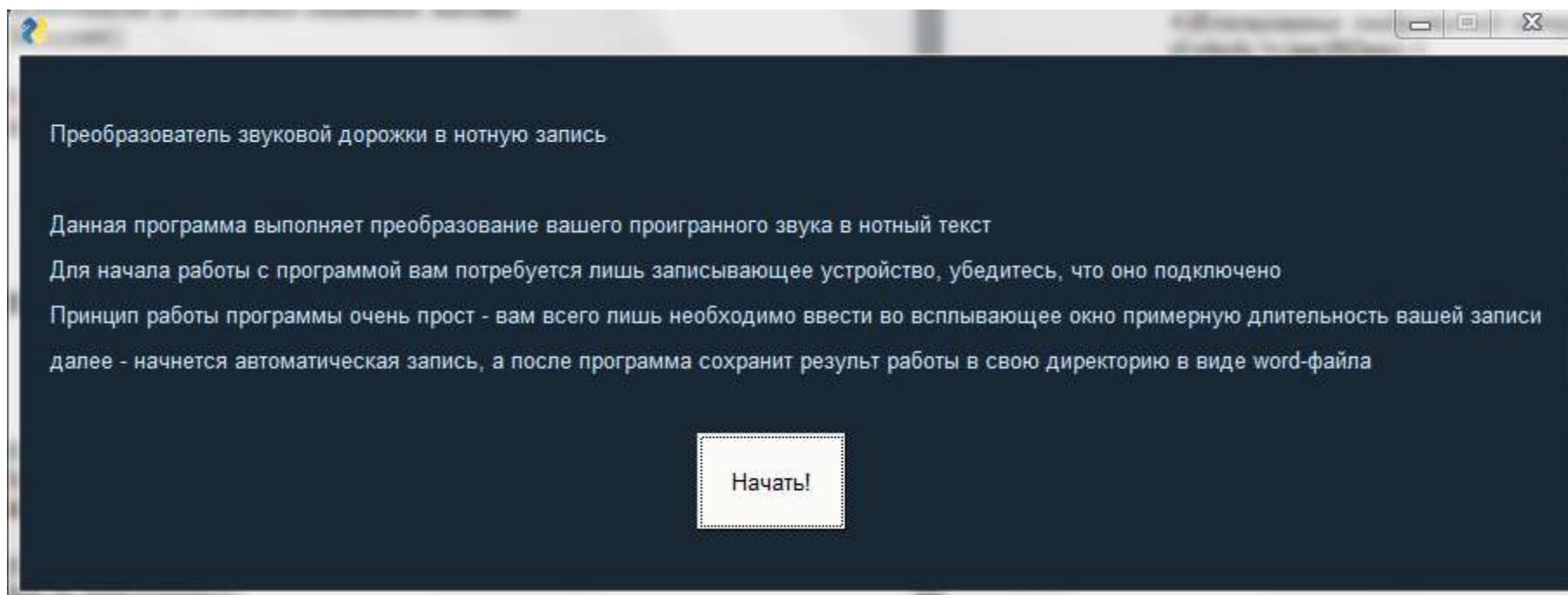
    p = document.add_paragraph()
    r = p.add_run()

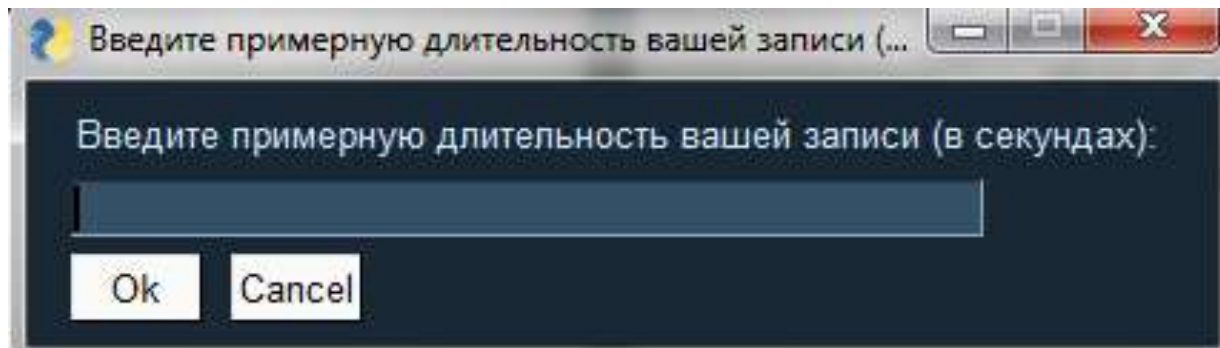
    r.add_picture('pics/sk.png') #Скрипичный ключ



    i = 1
    while i < len(arr):
        if arr[i] == 'A': r.add_picture('pics/a.png')
        elif arr[i] == 'A#': r.add_picture('pics/a1.png')
        elif arr[i] == 'B': r.add_picture('pics/b.png')
        elif arr[i] == 'C': r.add_picture('pics/c.png')
        elif arr[i] == 'C#': r.add_picture('pics/c1.png')
        elif arr[i] == 'D': r.add_picture('pics/d.png')
        elif arr[i] == 'D#': r.add_picture('pics/d1.png')
        elif arr[i] == 'E': r.add_picture('pics/e.png')
        elif arr[i] == 'F': r.add_picture('pics/f.png')
        elif arr[i] == 'F#': r.add_picture('pics/f1.png')
        elif arr[i] == 'G': r.add_picture('pics/g.png')
        elif arr[i] == 'G#': r.add_picture('pics/g1.png')
        i += 1

    document.save('Результат программы.docx')
```

Принцип работы программы





 project_v1.0.exe	04.11.2020 13:12	Приложение	29 765 КБ
 Результат программы.docx	04.11.2020 19:27	Документ Micros...	42 КБ



SONATE

Sonata quasi una Fantasia
Der Gräfin Julie Guicciardi gewidmet

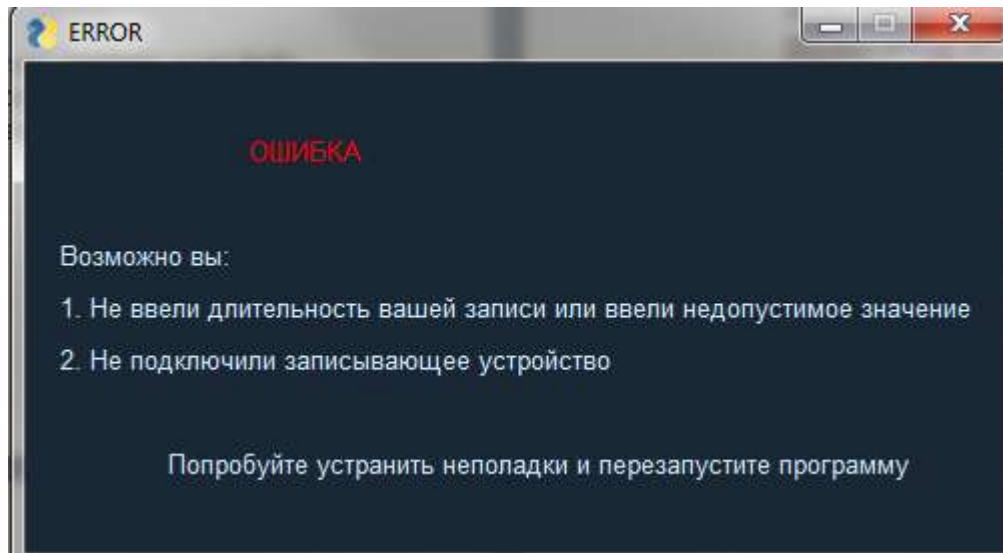
Adagio sostenuto

Si deve suonare tutto questo pezzo delicatissimamente

I., van BEETHOVEN, Op. 27, No. 2.

sempre pp

pp[ma cantabile]



```
def error_win(): # Функция ошибок - содержит варианты всех неправильных действий пользователя.
    sg.theme('DarkBlue')
    layout = [
        [sg.Text(' ')],
        [sg.Text(' ' * 20), sg.Text('ОШИБКА', text_color='red')],
        [sg.Text(' ')],
        [sg.Text('Возможно вы: ')],
        [sg.Text('1. Не ввели длительность вашей записи или ввели недопустимое значение')],
        [sg.Text('2. Не подключили записывающее устройство')],
        [sg.Text(' ')],
        [sg.Text(' ' * 10), sg.Text('Попробуйте устранить неполадки и перезапустите программу')],
        [sg.Text(' ')],
    ]
    window = sg.Window('ERROR', layout).Finalize()
    while True:
        event, values = window.read()
        if event in (None, 'Exit'):
            break
```