Здесь можно увидеть, что приложение продолжает работать, даже после того, как главный поток завершился (сообщение: “App stop”).

Для того, чтобы потоки не мешали остановке приложения (т.е. чтобы они останавливались вместе с завершением работы программы) необходимо при создании объекта *Thread* аргументу *daemon* присвоить значение *True*, либо после создания потока, перед его запуском присвоить свойству *daemon* значение *True*. Изменим процесс создания потока в приведенной выше программе:

```
th = Thread (target=func, daemon=True)
```

Запустим ее, получим следующий результат:

```
from child thread: 0
```

```
App stop
```

Поток остановился вместе с остановкой приложения.

1. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс. - Москва: Огни, 2013. - 177 с.
2. Бэрри, Пол Изучаем программирование на Python / Пол Бэрри. - М.: Эксмо, 2016. - 332 с.
3. Васильев, А. Н. Python на примерах. Практический курс по программированию / А.Н. Васильев. - М.: Наука и техника, 2016. - 432 с.
4. Васильев, Александр Николаевич Python на примерах. Практический курс по программированию. Руководство / Васильев Александр Николаевич. - М.: Наука и техника, 2017. - 752 с.
5. Гуриков, С.Р. Основы алгоритмизации и программирования на Python / С.Р. Гуриков. - М.: Форум, 2018. - 991 с.
6. Гуриков, С.Р. Основы алгоритмизации и программирования на Python. Учебное пособие. Гриф МО РФ / С.Р. Гуриков. - М.: Инфра-М, Форум, 2018. - 707 с.

Шавтикова Л.М., Текеев М.Б.

Создание интерфейса к задаче вычислительной математики с использованием графики

Северо-Кавказская государственная академия
(Россия, Черкесск)

doi: 10.18411/lj-12-2020-47

idsp: ljournal-12-2020-47

Аннотация

В статье рассматривается создание интерфейса к задаче вычислительной математики с использованием библиотек для рисования графики на языке программирования Python.

Ключевые слова: программирование, 2D-графика, численный анализ, фреймворк, GUI приложение.

Abstract

The article discusses the creation of an interface to a computational mathematics problem using libraries for drawing graphics in the Python programming language.

Keywords: programming, 2D graphics, numerical analysis, framework, GUI application.

В Python есть довольно много GUI фреймворков (graphical user interface), однако только Tkinter встроен в стандартную библиотеку языка. У Tkinter есть несколько преимуществ. Он кроссплатформенный, поэтому один и тот же код можно использовать на Windows, macOS и Linux. Визуальные элементы отображаются через

собственные элементы текущей операционной системы, поэтому приложения, созданные с помощью Tkinter, выглядят так, как будто они принадлежат той платформе, на которой они работают. Главным элементом GUI Tkinter является окно. Окнами называют контейнеры, в которых находятся все GUI элементы. Данные GUI элементы, к числу которых относятся текстовые боксы, ярлыки и кнопки, называются виджетами. Виджеты помещаются внутри окон.

Matplotlib - это библиотека Python, используемая для создания 2D-графиков и графиков с использованием сценариев Python. Он имеет модуль с именем pyplot, который упрощает построение графиков, предоставляя возможность управления стилями линий, свойствами шрифтов, осями форматирования и т. д. Он поддерживает очень широкий спектр графиков и графиков, а именно - гистограммы, гистограммы, спектры мощности, графики ошибок и т. д. Он используется вместе с NumPy для предоставления среды, которая является эффективной альтернативой с открытым исходным кодом для MatLab. Его также можно использовать с графическими инструментами, такими как PyQt и wxPython.

NumPy это open-source модуль для python, который предоставляет общие математические и числовые операции в виде пре-скомпилированных, быстрых функций. Они объединяются в высокоуровневые пакеты. Они обеспечивают функционал, который можно сравнить с функционалом MatLab. NumPy (Numeric Python) предоставляет базовые методы для манипуляции с большими массивами и матрицами. SciPy (Scientific Python) расширяет функционал numpy огромной коллекцией полезных алгоритмов, таких как минимизация, преобразование Фурье, регрессия, и другие прикладные математические техники.

Листинг:

```
import tkinter as tk
from tkinter import ttk
import matplotlib.pyplot as plt
import numpy as np
from bisection import bisection
from golden import golden
def bisection_tab():
    def f(x):
        #(1. + x**2) * np.exp(-x) + np.sin(x)
        return eval(name2.get())
    x = np.linspace(0., 10., 200)
    y = f(x)
    plt.plot(x, y)
    plt.xlabel('x')
    plt.grid(True)
    #plt.show()
    xRoot1 = bisection(f, 2, 4)
    #print('root(1) = ', xRoot1)
    xRoot2 = bisection(f, 6, 8)
    #print('root(2) = ', xRoot2)
    xRoot3 = bisection(f, 8, 10)
    #print('root(3) = ', xRoot3)
    ttk.Label(frame2, text='root(1) = '+str(xRoot1)).grid(column=0, row=0)
    ttk.Label(frame2, text='root(2) = '+str(xRoot2)).grid(column=0, row=1)
    ttk.Label(frame2, text='root(3) = '+str(xRoot3)).grid(column=0, row=2)
    plt.show()
```

```

def seidel_tab():
    from seidel import seidel
    n = 10
    al = 0.5
    A = np.zeros((n, n), 'float')
    for i in range(0, n):
        A[i,i] = 2
        if i > 0:
            A[i, i-1] = - 1 + al
        if i < n - 1:
            A[i, i+1] = - 1 - al
    #print('A:\n', A)
    f = np.zeros((n), 'float')
    f[0] = 1 - al
    f[n-1] = 1 + al
    #print('f:\n', f)
    x, iter = seidel(A, f)
    #print('iter = ', iter)
    #print('x:\n', x)
    ttk.Label(frame1, text='A:\n' + str(A)).grid(column=0, row=0)
    ttk.Label(frame1, text='f:\n' + str(f)).grid(column=0, row=1)
    ttk.Label(frame1, text='iter = ' + str(iter)).grid(column=0, row=2)
    ttk.Label(frame1, text='x:\n' + str(x)).grid(column=0, row=3)
def golden_tab():
    def f(x):
        #return (x**2 - 6.*x + 12.) / (x**2 + 6.*x + 12.)
        return eval(name3.get())
    a = 0.
    b = 20.
    x = np.linspace(a, b, 200)
    y = f(x)
    plt.plot(x, y)
    plt.xlabel('x')
    plt.grid(True)
    #plt.show()
    xMin, fMin = golden(f, a, b)
    #print('xMin=', xMin)
    #print('xMin=', fMin)
    ttk.Label(frame3, text='xMin=' + str(xMin)).grid(column=0, row=0)
    ttk.Label(frame3, text='xMin=' + str(fMin)).grid(column=0, row=1)
    plt.show()
def interpolation_tab():
    from interpolation import interpolation, coef
    def f(x):
        #return 1. / (1 + 25*x**2)
        return eval(name5.get())
    a = -1.
    b = 1.
    x1 = np.linspace(a, b, 200)
    y1 = f(x1)

```

```
plt.plot(x1, y1)
nList = [4, 6, 10]
sgList = ['--', '-.', ':']
for k in range(len(nList)):
    n = nList[k]
    x = np.linspace(a, b, n+1)
    y = f(x)
    plt.scatter(x, y, marker='o')
    c = coef(x, y)
    y1 = interpolation(c, x, x1)
    sl = 'n=' + str(n)
    sg = sgList[k]
    plt.plot(x1, y1, sg, label=sl)
plt.xlabel('x')
plt.grid(True)
plt.legend(loc=0)
plt.show()
root = tk.Tk()
root.title("Численный Анализ")
tabControl = ttk.Notebook()
tab1 = ttk.Frame(tabControl)
tabControl.add(tab1, text='Метод Зейделя')
name1 = tk.StringVar()
name1Entered = ttk.Entry(tab1, width=12, textvariable=name1)
name1Entered.grid(column=0, row=1, stick=tk.W)
frame1 = ttk.LabelFrame(tab1, text='Результат вычислений')
frame1.grid(column=0, row=2)
action1 = ttk.Button(tab1, text="Click Me!", command=seidel_tab)
action1.grid(column=0, row=1, padx=80, pady=4)
tab2 = ttk.Frame(tabControl)
tabControl.add(tab2, text='Метод бисекций') (Рисунок 1)
tabControl.pack(expand=1, fill="both")
name2 = tk.StringVar()
name2Entered = ttk.Entry(tab2, width=40, textvariable=name2)
name2Entered.grid(column=0, row=0, stick=tk.W)
frame2 = ttk.LabelFrame(tab2, text='Результат вычислений')
frame2.grid(column=0, row=1)
action2 = ttk.Button(tab2, text="Click Me!", command=bisection_tab)
action2.grid(column=1, row=0)
tab3 = ttk.Frame(tabControl)
tabControl.add(tab3, text='Метод золотого сечения') (Рисунок 2)
tabControl.pack()
name3 = tk.StringVar()
name3Entered = ttk.Entry(tab3, width=40, textvariable=name3)
name3Entered.grid(column=0, row=1, stick=tk.W)
frame3 = ttk.LabelFrame(tab3, text='Результат вычислений')
frame3.grid(column=0, row=2)
action3 = ttk.Button(tab3, text="Click Me!", command=golden_tab)
action3.grid(column=2, row=1)
tab4 = ttk.Frame(tabControl)
```

```

tabControl.add(tab4, text='Метод Рунге-Куты')
tabControl.pack()
tab5 = ttk.Frame(tabControl)
tabControl.add(tab5, text='Метод Интерполяции') (Рисунок 3)
tabControl.pack()
name5 = tk.StringVar()
name5Entered = ttk.Entry(tab5, width=12, textvariable=name5)
name5Entered.grid(column=0, row=1, stick=tk.W)
action5 = ttk.Button(tab5, text="Click Me!", command=interpolation_tab)
action5.grid(column=2, row=1)
root.mainloop()

```

Рисунок 1 - Метод бисекций

Рисунок 2 - Метод золотого сечения

Рисунок 3 - Метод интерполяции

1. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс. - Москва: Огни, 2013. - 177 с.
2. Бэрри, Пол Изучаем программирование на Python / Пол Бэрри. - М.: Эксмо, 2016. - 332 с.
3. Васильев, А. Н. Python на примерах. Практический курс по программированию / А.Н. Васильев. - М.: Наука и техника, 2016. - 432 с.
4. Васильев, Александр Николаевич Python на примерах. Практический курс по программированию. Руководство / Васильев Александр Николаевич. - М.: Наука и техника, 2017. - 752 с.
5. Гуриков, С.Р. Основы алгоритмизации и программирования на Python / С.Р. Гуриков. - М.: Форум, 2018. - 991 с.
6. Гуриков, С.Р. Основы алгоритмизации и программирования на Python. Учебное пособие. Гриф МО РФ / С.Р. Гуриков. - М.: Инфра-М, Форум, 2018. - 707 с.

Элиханов В.Г., Матыгов М.М.

Как оптимизировать сайт для увеличения конверсии

Чеченский государственный университет
(Россия, Грозный)

doi: 10.18411/lj-12-2020-48

idsp: ljjournal-12-2020-48

Аннотация

В этой работе рассматривается актуальная тема оптимизации веб-сайта и увеличения конверсии. Обсудим, какие действия можно предпринять для достижения более высоких позиций и оптимизации производительности целевой страницы. Также в данной статье даны общие рекомендации по продвижению сайта.

Ключевые слова: увеличение конверсии сайта, оптимизация, seo, сайт.

Abstract

The article describes the hot topic of website optimization and conversions. We will discuss what actions can be taken to achieve higher positions and optimize the performance of the achievement goal. Also, this article provides general recommendations for website promotion.

Keywords: increase site conversion, optimization, seo, site.