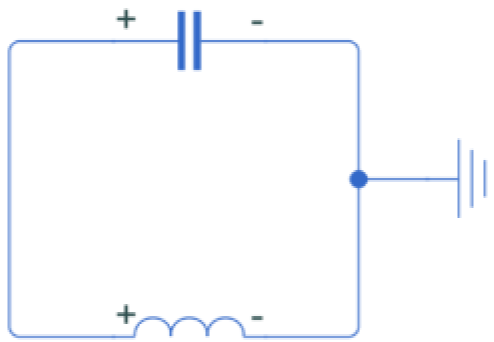
Дана электрическая схема, состоящая из идеального конденсатора ёмкостью C = 1 Ф, идеального индуктора индуктивностью L = 1 Гн и заземления. Пусть в начальный момент времени конденсатор заряжен до 1 В, а ток в цепи отсутствует.



Постройте соответствующую электрической схеме задачу Коши для системы обыкновенных дифференциальных уравнений и напишите программу для её решения любым численным методом на интервале $t \in [0; 100]$.

Приведите графики изменения тока и напряжения конденсатора. Обоснуйте правильность полученного решения как с точки зрения физики процесса, так и с точки зрения математики.

Решение.

Будем считать заряд конденсатора q положительным, если знаки заряда на обкладка совпадают с показанными на рисунке, а силу тока положительной, если ток направлен против часовой стрелки. Согласно закону Ома, в замкнутом контуре сумма падений напряжения равна сумме действующих ЭДС. На конденсаторе – напряжение $u = q/C$, на катушке наводится самоиндукция $\varepsilon = -L\, di/dt$, в точке заземление нет ни падения напряжения, ни ЭДС (т.е. заземление не влияет на колебательный контур).

$$u = \varepsilon.$$

Поскольку $i = -\,dq/dt = -Cu'$, имеем

$$u = -LCu''$$

или

$$u'' + \frac{1}{LC}u = 0$$

с начальными условиями $u(0) = u_0$, $i(0) = -Cu' = 0$.

Аналитическое решение этой задачи хорошо известно:

$$u(t) = u_0 \cos \omega t,$$

$$i(t) = u_0 \omega C \sin \omega t,$$

где $\omega = 1/\sqrt{LC}$ – собственная частота колебаний.

Для численного решения исходный диффур второго порядка $u'' + \omega^2 u = 0$ будем сводить к системе 2-х дифференциальных уравнений первого порядка через переобозначения $u_0 = u$, $u_1 = u'$. Система примет вид

$$\begin{cases} u_0' = u_1 \\ u_1' = -\omega^2 u_0 \end{cases}$$

с соответствующим вектором начальных условий $\{u_0, u_1\}_0 = \{1, 0\}$. Каждое уравнение системы $u_j' = f_j(t, u_0, u_1)$, $j = \overline{0, 1}$, переписывается через разностною схему.

В качестве основного решателя выбрана схема Рунге-Кутты 4-го порядка как самая популярная.

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + \frac{h}{6}(\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4).$$

На одном шаге вектор-функция вычисляется 4 раза:

$$\boldsymbol{k}_1 = \boldsymbol{f}(x_n, \boldsymbol{y}_n),$$

$$\boldsymbol{k}_2 = \boldsymbol{f}\left(x_n + \frac{h}{2}, \boldsymbol{y}_n + \frac{h}{2}\boldsymbol{k}_1\right),$$

$$\boldsymbol{k}_3 = \boldsymbol{f}\left(x_n + \frac{h}{2}, \boldsymbol{y}_n + \frac{h}{2}\boldsymbol{k}_2\right),$$

$$\boldsymbol{k}_4 = \boldsymbol{f}(x_n + h, \boldsymbol{y}_n + h\boldsymbol{k}_3).$$

У метода 4-й порядок точности: суммарная ошибка имеет порядок $O(h^4)$.

Программировать будем на Python 3. Нам потребуется пакеты для работы с массивами numpy и рисования графиков matplotlib:

```
import numpy as np
import matplotlib.pyplot as plt
```

Реализуем функцию runge_kutta_4(f, T, U0), которая возвращает сетку решений $\boldsymbol{u}_i$. Входные параметры функции:

- f – вектор-функция правых частей системы,
- T – вектор с разбиением по независимой переменной
- U0 – вектор начальных значений $\{u_0, u_1\}_0$,

Программный код метода-функции runge_kutta_4(f, T, U0) представлен ниже.

```
def runge_kutta_4(f, T, U0):
    U = np.zeros((len(T), len(U0)))
    U[0] = U0

    h = T[1] - T[0]

    for i in range(1, len(T)):
        k1 = f(T[i - 1], U[i - 1])
        k2 = f(T[i - 1] + h / 2, U[i - 1] + h * k1 / 2)
        k3 = f(T[i - 1] + h / 2, U[i - 1] + h * k2 / 2)
        k4 = f(T[i - 1] + h, U[i - 1] + h * k3)

        U[i] = U[i - 1] + h * (k1 + 2 * k2 + 2 * k3 + k4) / 6

    return U
```

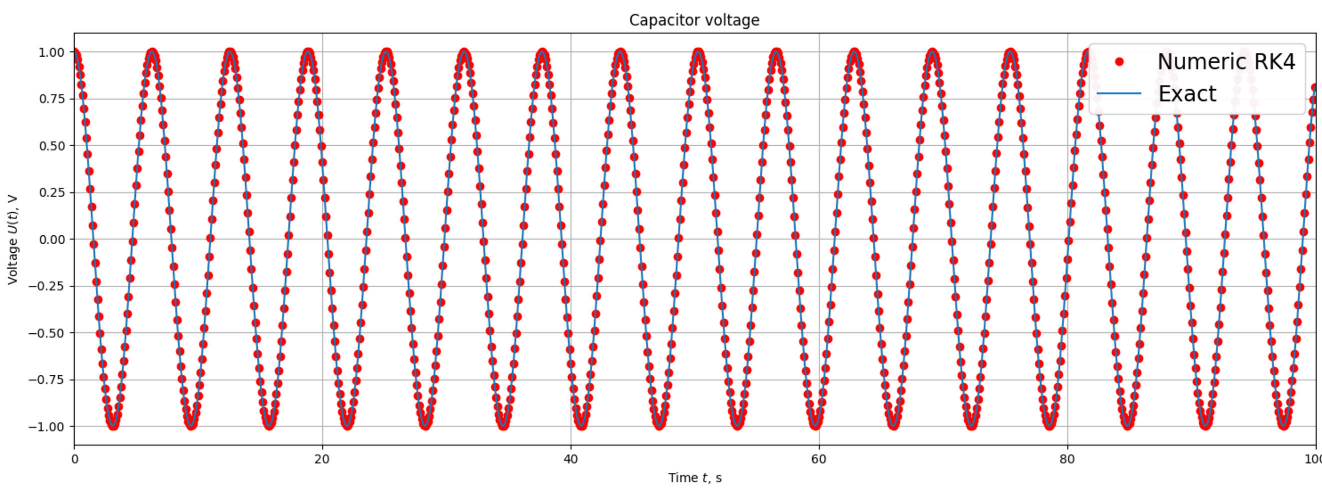
Массив функции правых частей системы $f_j(t, u_0, u_1)$:

```
def F(t, U):
    """Right hand side function"""

    f = np.zeros(U.shape)
    f[0] = U[1]
    f[1] = (-1) * omega ** 2 * U[0]

    return f
```

Результаты работы программы (т.е. решение исходной задачи Коши) представлен на графиках.

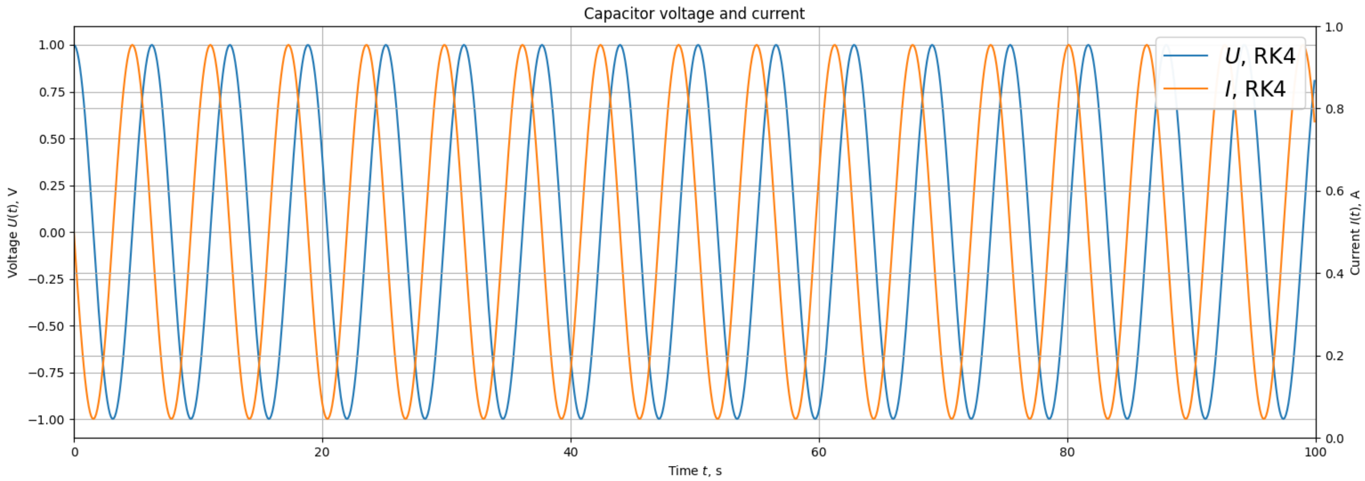
1. Напряжение на конденсаторе.

Это функция u_0 из системы или нулевой столбец ответного массива U[:,0]. Шаг по времени выбран $h = 0{,}1$ с. Все точки численного решения нанесены на график в виде красных кружков. Визуально они прекрасно накладываются на точное аналитическое решение (сплошная синяя линия).



2. Ток в цепи.

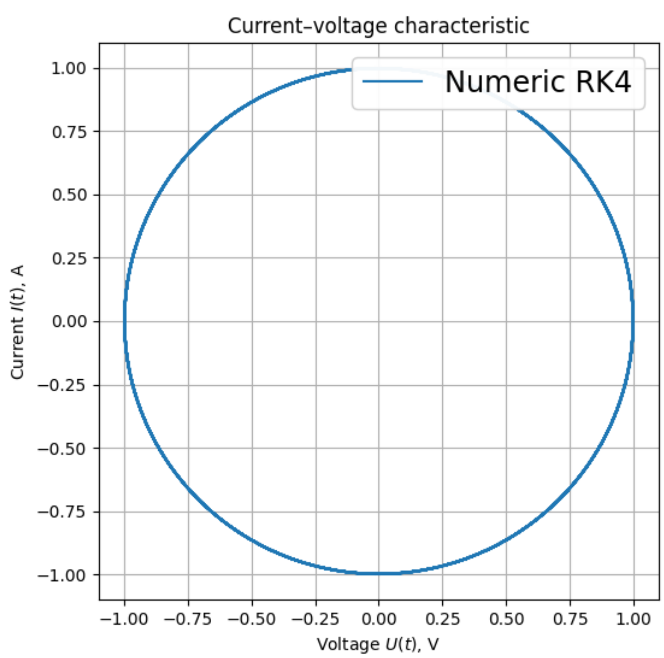
Это функция u_1 из системы или первый столбец ответного массива U[:,1] с размерным множителем ωC. Зависимость силы тока в цепи представлена совместно с напряжением на конденсаторе, чтобы наглядно продемонстрировать смещение фаз между током и напряжением в четверть периода. Шаг по времени выбран $h = 0{,}1$ с. Желтая линия показывает силу тока, полученную численно методом Рунге-Кутты 4 порядка; синяя линия показывает напряжение на конденсаторе, полученное в этом же численном решении.



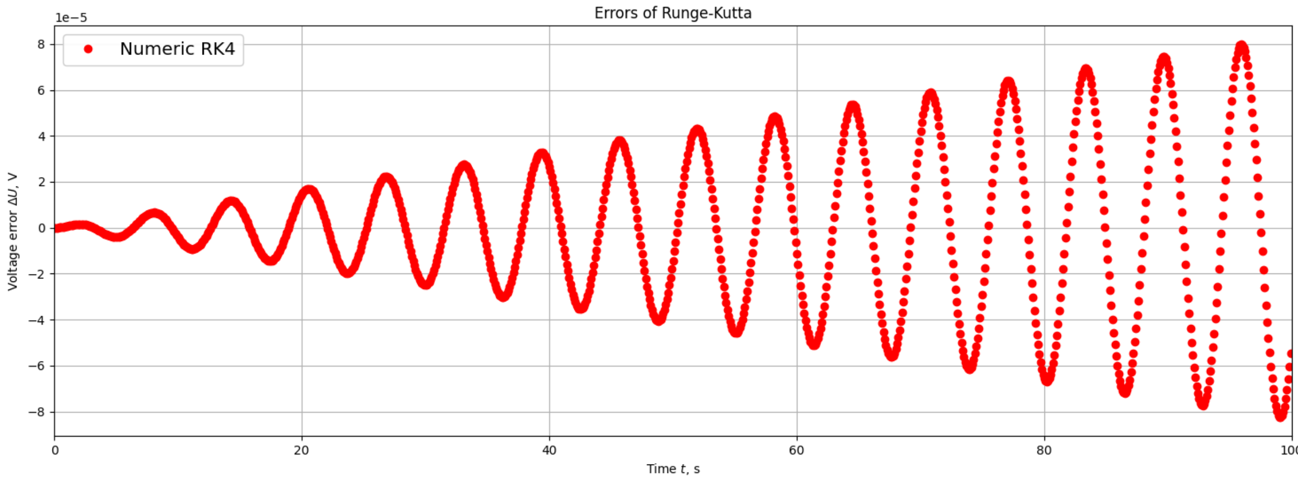
3. Ошибки.

Так как известно точное аналитическое решение, было бы интересно с ним свериться.

Если численное решение стабильно, то на совместном графике (u', u) будет замкнутая кривая (эллипс), т.к. $u \sim \cos \omega t$, $i \sim \sin \omega t$. На графике численное решение, шаг $h = 0{,}1$ с.



Тем не менее, если посмотреть на разницу численного решения и идеального, то видно, что ошибка накапливается и через $\sim 10^4$ интервалов станет сравнима с величиной самого решения. На графике ошибки численного решения для u_0, шаг $h = 0{,}1$ с.



В рамках данного интервала $t \in [0; 100]$ ошибка согласуется с ожиданием $O(h^4)$. Если требуется повысить точность, то самым простым способом будет уменьшение шага разбиения, но это будет сильно влиять на сложность вычислений.

Функция runge_kutta_4 f, T, U0) находится в модуле models.py. В файле main.py происходит инициализация задачи, т.е. задание задачи Коши, вызов решателей и визуализация решения.